NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY, GREATER NOIDA,GAUTAM BUDDH NAGAR

(An Autonomous Institute)

NIET
Greater Noida
An Autonomous Institute
Estd.2001
Get Future Ready

Affiliated to

DR. A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY, UTTAR PRADESH LUCKNOW

Evaluation Scheme & Syllabus

For

B.Tech in Computer Science & Business Systems (CSBS)

Second Year

(Effective from the Session: 2026-27)

NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY, GREATER NOIDA
(An Autonomous Institute)

B. TECH (CSBS)
Evaluation Scheme
SEMESTER III

Sl. No.	Subject Codes	Subject Name	Types of Subjects	Periods			Evaluation Schemes				End Semester		Total	Credit
				L	T	P	CT	TA	TOTAL	PS	TE	PE		
1 WEEKS COMPULSORY INDUCTION PROGRAM														
1	CCSBS0303	Formal Language and Automata Theory	Mandatory	3	1	0	30	20	50		100		150	4
2	CCSBS0301	Computer Organization & Architecture	Mandatory	3	0	0	30	20	50		100		150	3
3	CCSBS0304	Object Oriented Programming	Mandatory	2	0	0	30	20	50		50		100	2
4	CCSBS0302	Computational Statistic	Mandatory	3	0	0	30	20	50		100		150	3
5	CCSBS0305	Software Engineering	Mandatory	2	0	0	30	20	50		50		100	2
6	CCSBS0351	Computer Organization & Architecture Lab	Mandatory	0	0	2				25		25	50	1
7	CCSBS0354	Object Oriented Programming Lab	Mandatory	0	0	4				50		50	100	2
8	CCSBS0352	Computational Statistic Lab	Mandatory	0	0	2				25		25	50	1
9	CCSBS0355	Software Engineering Lab	Mandatory	0	0	4				50		50	100	2
10	CNC0303	Indian Constitution	Mandatory	2	0	0	30	20	50			50	100	NA
		Massive Open Online Courses (For B.Tech. Hons. Degree)	*MOOCs											
		TOTAL		**13**	**1**	**12**	**180**	**120**	**250**	**150**	**400**	**200**	**950**	**20**

*** List of MOOCs Based Recommended Courses for Second year B. Tech Students**

S. No.	Subject Code	Course Name	University / Industry Partner Name	No of Hours	Credits
1	CMC0031	Gen AI 101	NASSCOM	15 hrs	1
2	CMC0032	Python for Data Science	NASSCOM	30h	2
3	CMC0033	Software Engineering Fundamentals - Software Development and Testing	Infosys Wingspan (Infosys Springboard)	16 h 56min	1
4	CMC0034	Computational Problem Solving	Infosys Wingspan (Infosys Springboard)	27h 26m	2

PLEASE NOTE: -

- **A 3–4-week Internship shall be conducted during summer break after semester-II and will be assessed during semester-III.**
- **Compulsory Audit (CA) Courses (Non-Credit - CNC0303)**
  - All Compulsory Audit Courses (a qualifying exam) do not require any credit.
  - The total and obtained marks are not added to the grand total.

Abbreviation Used:

L: Lecture, T: Tutorial, P: Practical, CT: Class Test, TA: Teacher Assessment, PS: Practical Sessional, TE: Theory End Semester Exam., PE: Practical End Semester Exam, CE: Core Elective, OE: Open Elective, DE: Departmental Elective, CA: Compulsory Audit, MOOCs: Massive Open Online Courses.

NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY, GREATER NOIDA
(An Autonomous Institute)

B. TECH (CSBS)
Evaluation Scheme
SEMESTER IV

Sl. No.	Subject Codes	Subject Name	Types of Subjects	Periods			Evaluation Schemes				End Semester		Total	Credit
				L	T	P	CT	TA	TOTAL	PS	TE	PE		
1	CCSBS0403	Operating Systems	Mandatory	3	0	0	30	20	50		100		150	3
2	CCSBS0401	Database Management Systems	Mandatory	3	0	0	30	20	50		100		150	3
3	CCSBS0404	Software Design with UML	Mandatory	3	0	0	30	20	50		100		150	3
4	CCSBS0406	Introduction to Innovation, IP Management & Entrepreneurship	Mandatory	3	0	0	30	20	50		100		150	3
5	CCSBS0402	Operations Research	Mandatory	2	0	0	30	20	50		50		100	2
6	CCSBS0405	Design Thinking	Mandatory	2	0	0	30	20	50		50		100	2
7	CCSBS0453	Operating Systems Lab (Unix)	Mandatory	0	0	2				25		25	50	1
8	CCSBS0451	Database Management Systems Lab	Mandatory	0	0	2				25		25	50	1
9	CCSBS0454	Software Design with UML Lab	Mandatory	0	0	2				25		25	50	1
10	CCSBS0452	Operations Research Lab	Mandatory	0	0	2				25		25	50	1
11	CCSBS0455	Design Thinking Lab	Mandatory	0	0	2				25		25	50	1
12	CNC0404	Essence of Indian Traditional Knowledge	Mandatory	2	0	0	30	20	50		50		100	NA
		Massive Open Online Courses (For B.Tech. Hons. Degree)	*MOOCs											
		Total		**16**	**0**	**10**	**210**	**140**	**300**	**125**	**550**	**125**	**1050**	**21**

*** List of MOOCs Based Recommended Courses for Second year B. Tech Students**

S. No.	Subject Code	Course Name	University/ Industry Partner Name	N. of Hours	Credits
1	CMC0044	Getting Started with Linux Fundamentals 9.1	RedHat	16h	1
2	CMC0055	RHEL Automation with Ansible 9.0	RedHat	32hr	2
3	CMC0056	Implementing Database Using SQL Server 2012	Infosys Wingspan (Infosys Springboard)	26h 53m	2
4	CMC0057	SQL Fundamentals	NASSCOM	8h	0.5

PLEASE NOTE: -

- **Compulsory Audit (CA) Courses (Non-Credit - CNC0404)**
  - All Compulsory Audit Courses (a qualifying exam) do not require any credit.
  - The total and obtained marks are not added to the grand total.

Abbreviation Used:

L: Lecture, T: Tutorial, P: Practical, CT: Class Test, TA: Teacher Assessment, PS: Practical Sessional, TE: Theory End Semester Exam., PE: Practical End Semester Exam, CE: Core Elective, OE: Open Elective, DE: Departmental Elective, CA: Compulsory Audit, MOOCs:Massive Open Online Courses.

NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY, GREATER NOIDA
(An Autonomous Institute)

AICTE Guidelines in Model Curriculum:

A student will be eligible to get Under Graduate degree with Honours only, if he/she completes the additional MOOCs courses such as Infosys Springboard/NASSCOM/CISCO (as applicable) certifications, or any other online courses recommended by the Institute (Equivalent to 20 credits). During Complete B.Tech. Program Guidelines for credit calculations are as follows.

1. For 6 to 12 Hours =0.5 Credit
2. For 13 to18 =1 Credit
3. For 19 to 24 =1.5 Credit
4. For 25 to 30 =2 Credit
5. For 31 to 35 =2.5 Credit
6. For 36 to 41 =3 Credit
7. For 42 to 47 =3.5 Credit
8. For 48 and above =4 Credit

For registration to MOOCs Courses, the students shall follow Infosys Springboard/NASSCOM/CISCO (as applicable) registration details as per the assigned login and password by the Institute these courses may be cleared during the B. Tech degree program (as per the list provided). After successful completion of these MOOCs courses, the students shall provide their successful completion status/certificates to the Controller of Examination (COE) of the Institute through their coordinators/Mentors only.

The students shall be awarded Honours Degree as per following criterion.

i. If he / she secures 7.50 as above CGPA.
ii. Passed each subject of that degree program in the single attempt without any grace.
iii. Successful completion of MOOCs based 20 credits

Course Code: CCSBS0303	Course Name: Formal Language and Automata Theory	L	T	P	C
Course Offered in: B.Tech (CSBS)		3	1	0	4

Pre-requisite: The student should have basic knowledge of discrete mathematics and Fundamental of Computer System

Course Objectives: The objective of the course is to present an introduction to mathematical foundations of computation including automata theory, provide the design concepts of abstract computation model of finite automata, push down automata and Turing Machine and familiarize the notions of algorithm, decidability, complexity, and computability.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Understand the concepts of formal languages, grammars and design finite automata for language recognition	K2,K3
CO2	Analyze regular languages using regular expressions, regular grammars, and their properties.	K4
CO3	Construct context-free grammars and pushdown automata for context-free languages.	K3
CO4	Analyze Turing machines and their computational models for language recognition.	K4
CO5	Analyze undecidability and computational complexity using reductions and complexity classes.	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO 7	PO8	PO9	PO1 0	PO11	PSO1	PSO2	PSO3
CO1	3	2	3	-	1	-	-	-	-	-	1	3	2	1
CO2	3	3	2	2	1	-	-	-	-	-	1	3	2	1
CO3	3	2	3	2	1	-	-	-	-	-	1	3	3	1
CO4	3	3	2	3	1	-	-	-	-	-	2	3	2	2
CO5	3	3	1	3	-	-	-	-	-	-	2	3	2	2

Course Contents / Syllabus

UNIT- 1	Introduction Of Finite Automata	12 hours

Alphabet,Strings , languages and grammars, productions and derivation, Chomsky hierarchy of languages. Introduction to Finite State Machine, Deterministic finite automata (DFA) , Nondeterministic finite automata (NFA) and equivalence with DFA, NFA with ∈-Transition, Equivalence of NFA's with and without ∈-Transition, Minimization of Finite Automata

UNIT - 2	Regular Language and Finite Automata	15 hours

Regular expressions and languages, Properties of Regular Expression , Finite Automata and Regular Expression, Arden's theorem, Regular grammars and equivalence with finite automata, properties of regular languages, Kleene's theorem, pumping lemma for regular languages, Myhill-Nerode theorem and its uses

UNIT - 3	Context-free Languages and Pushdown Automata	12 hours

Context-free grammars (CFG) and languages (CFL), Chomsky and Greibach Normal Forms, Equivalence with CFG, Parse trees, Ambiguity in CFG, Pumping lemma for Context-free languages, Deterministic Pushdown Automata, Nondeterministic Pushdown Automata (PDA), Closure Properties of CFLs. **Context-sensitive languages**: Context-sensitive grammars (CSG) and languages, Linear Bounded automata and Equivalence with CSG.		
UNIT- 4	**Turing machines**	**11 hours**
The basic model for Turing machines (TM), Turing recognizable (Recursively Enumerable) and Turing- decidable (recursive) Languages and their closure properties, Variants of Turing machines, Non deterministic TMs and Equivalence with Deterministic TMs, Unrestricted Grammars and Equivalence with Turing machines, TM as Enumerators.		
UNIT - 5	**Undecidability and Basic Introduction to Complexity**	**10 hours**
Church-Turing thesis, Universal Turing machine, Universal and diagonalization languages, Reduction between languages and Rice's theorem, Undecidable problems about languages Introductory ideas on Time complexity of deterministic and nondeterministic Turing machines, P and NP, NP-completeness, Cook's Theorem, other NP -Complete problems.		
Total Lecture Hours		**60 hours**
Textbook:		
S.No	**Book Title with publication agency & year**	**Author**
1.	Introduction to Automata Theory, Languages, and Computation	John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman.
S.No	**Book Title**	**Author**
1. 2. 3. 4. 5.	Elements of the Theory of Computation Introduction to Languages and the Theory of Computation Computers and Intractability: A Guide to the Theory of NP Completeness Automata and Computability Introduction to the Theory of Computation	Harry R. Lewis and Christos H.Papadimitriou John Martin M. R. Garey and D. S. Johnson Dexter C. Kozen
Self-Study Content Links:		Duration (in Hrs)
MOOCs	Computational Theory: Language Principle & Finite Automata Theory https://infyspringboard.onwingspan.com/web/en/app/search/learning?lang=en&q=finite%20automata&p=0&f=%7B%22contentType%22:%5B%22Course%22%5D%7D	44 min
Video Lecture	https://nptel.ac.in/courses/106104028 https://www.youtube.com/playlist?list=PLwi7ySzY5bVKn6gPAEHzWC0hQ0Q0nvsA6	42 Hrs 35 Hrs

Mode of Evaluation

CIE							**ESE**	**Total**
ST1	**ST2**	**ST3**	**TA1** **5**	**TA2** **5**	**TA3** **5**	**Attendance** **5**		
30			**20**				**100**	**150**

Course Code: CCSBS0301	Course Name: Computer Organization & Architecture	L	T	P	C
Course Offered in: B.Tech (CSBS)		**3**	**0**	**0**	**3**

Pre-requisite: Basics of Boolean logic, Instruction set, Memory Management, Pipelining and Parallel Processing

Course Objectives: Student will learn different types of organization, structures and functions of computer, to understand the data representation and computer arithmetic. They will understand the concept of control unit, memory organization, peripheral devices and pipelining.

Course Outcome: After completion of the course the student will be able to		Bloom's Knowledge Level (KL)
CO1	Identify the computer hardware and how software interacts with computer hardware.	K1
CO2	Demonstrate how to add and multiply integers and floating-point numbers using two's complement and IEEE floating point representation.	K3
CO3	Understand the principles and the implementation of computer arithmetic and Program using x86 instruction sets.	K2
CO4	Identify the memory technologies, input-output systems and evaluate the performance of memory system.	K5
CO5	State and compare properties of shared memory and distributed multiprocessor systems and cache coherency protocols.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	-	-	2	-	-	-	-	-	1	2	2	2
CO2	3	2	-	-	2	-	-	-	-	-	1	2	3	2
CO3	3	2	2	-	3	-	-	1	2	-	2	3	3	3
CO4	3	2	-	-	3	-	-	-	-	-	1	2	3	3
CO5	2	2	2	1	2	-	-	-	1	-	2	3	3	3

Course Contents / Syllabus

Unit 1	**Boolean logic, Functional blocks, Instruction Set**	**08 hours**

Revision of basics in Boolean logic and Combinational/Sequential Circuits.

Functional blocks of a computer: CPU, memory, input-output subsystems, control unit.

Instruction set architecture of a CPU: Registers, instruction execution cycle, RTL interpretation of instructions, addressing modes, instruction set. Outlining instruction sets of some common CPUs.

Unit 2	**Data representation, Computer arithmetic**	**10 hours**

Data representation: Signed number representation, fixed and floating point representations, character representation.

Computer arithmetic: Integer addition and subtraction, ripple carry adder, carry look-ahead adder, etc. multiplication – shift-and-add, Booth multiplier, carry save multiplier, etc. Division restoring and non-restoring techniques, floating point arithmetic, IEEE 754 format.

Unit 3	**x86 architecture, control unit design, Memory system design**	**08 hours**

Introduction to x86 architecture.

CPU control unit design: Hardwired and micro-programmed design approaches, design of a simple hypothetical CPU.

Memory system design: Semiconductor memory technologies, memory organization.

Unit 4	**Peripheral devices, Pipelining**	**10 hours**

Peripheral devices and their characteristics: Input-output subsystems, I/O device interface, I/O transfers – program controlled, interrupt driven and DMA, privileged and non-privileged instructions, software interrupts and exceptions. Programs and processes – role of interrupts in process state transitions, I/O device interfaces – SCII, USB

Pipelining: Basic concepts of pipelining, throughput and speedup, pipeline hazards.

Unit 5	**Parallel Processors, Memory organization**	**9 hours**

Parallel Processors: Introduction to parallel processors, Concurrent access to memory and cache coherency

Memory organization: Memory interleaving, concept of hierarchical memory organization, cache memory, cache size vs. block size, mapping functions, replacement algorithms, write policies.

Total Lecture Hours	**45 hours**

Textbook:

S.No	**Book Title**	**Author**
1.	Introduction to the Indian Constitution, 8th Edition, PHI Learning Pvt. Ltd	Brij Kishore Sharma

Reference Books:

S.No	**Book Title**	**Author**
1.	Computer Organization	M. Mano Carl Hamacher
2.	Computer System Architecture	McGraw-Hill Zvonko Vranesic

Self-Study Content Links:		Duration (in Hrs)
MOOCs	**https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01384248588788531216013_shared/overview** https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_0140053563611217927045/overview https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013986372039647232268/overview https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01384206289267097611335_shared/overview https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01389293003676876816/overview	30
Video Lectures	https://www.youtube.com/playlist?list=PLxCzCOWd7aiHMonh3G6QNKq53C6oNXGrX https://www.youtube.com/watch?v=Ol8D69VKX2k	15 hour
Web References	https://www.learncomputerscienceonline.com/computer-organization-and-architecture/	10 hours

Note: Self-study material may include assignments, seminars, micro projects, industrial visits, any other student activities, etc.

Mode of Evaluation

CIE							**ESE**	**Total**
ST1	**ST2**	**ST3**	**TA1 5**	**TA2 5**	**TA3 5**	**Attendance 5**		
30			**20**				**100**	**150**

LAB Course Code: CCSBS0351	LAB Course Name: Computer Organization & Architecture Lab	L	T	P	C
Course Offered in: B. Tech (CSBS)		0	0	2	1

Pre-requisite: Basics of Boolean logic, Instruction set, Memory Management, Pipelining and Parallel Processing

Course Objectives: Student will learn different types of organization, structures and functions of computer, to understand the data representation and computer arithmetic. They will understand the concept of control unit, memory organization, peripheral devices and pipelining.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Implement combinational and sequential circuits.	K3
CO2	Implement shift registers, Arithmetic Logic Units (ALUs), and control units.	K3
CO3	Analyze and implement input/output systems with internal registers.	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	3	2	3	-	-	-	-	-	2	2	3	2
CO2	3	2	3	2	3	-	-	-	-	-	2	2	3	3
CO3	3	3	3	3	3	-	-	-	-	-	2	3	3	2
CO4	-	-	-	-	-	-	-	-	-	-	-	-	-	2
CO5	-	-	-	-	-	-	-	-	-	-	-	-	-	2

List Of Practical's (Indicative & Not Limited To)

Sr. No.	Program	CO, K level
1.	Design the control unit of a computer using hardwiring based on its RTL description.	CO1, K3
2.	Write a program to control an LED or 7-segment display. The program should turn the LED on and off or display digits based on user input.	CO1, K3
3.	Implementation of Half subtractor and full subtractor	CO2, K3
4.	Implementation of Half subtractor and full subtractor	CO1, K3
5.	Implementation of array multiplier.	CO1, K3
6.	Implementation of array multiplexer and demultiplexer.	CO1, K3
7.	Implementation of array encoder and decoder.	CO1, K3
8.	Implementation of Synchronous and Asynchronous counter.	CO2, K3
9.	Implementation of Shift registers.	CO2, K3
10.	Design of an arithmetic and logic unit	CO2, K3
11.	Design of an 8-bit input/output system with four 8-bit internal register.	CO2, K3
12.	Design the data path of a computer from its registers transfer language	CO2, K3

13.	Write an assembly or C program to read data from a keyboard or another input device using program-controlled I/O.	CO2, K3
14.	Write a program that configures a microprocessor or microcontroller to use interrupts for I/O operations. The program should demonstrate the handling of an interrupt when data is received from an input device.	CO2, K3
15.	Write a program to communicate with a USB device, such as a flash drive or keyboard, to read or write data.	CO2, K3
16.	Write a program to simulate a basic instruction pipeline with 4 stages: Fetch, Decode, Execute, and Write Back.	CO3, K4
17.	Write a program that simulates a pipeline processor and introduces various hazards like data hazards, control hazards, and structural hazards. Implement techniques such as forwarding, stalling, and branch prediction to resolve these hazards.	CO3, K4
18.	Write a program in a parallel programming language (e.g., OpenMP or MPI) to perform matrix multiplication using multiple processors.	CO3, K4
19.	Write a simulation program to model a multi-processor system where each processor has its own cache.	CO3, K4
20.	Implement a cache coherence protocol (e.g., MESI) to ensure data consistency across all caches when multiple processors access shared memory.	CO3, K4
21.	Write a program that simulates a multi-core processor system where multiple cores try to access and modify shared memory simultaneously. Implement synchronization techniques such as locks or semaphores to prevent race conditions.	CO3, K4

Total Hours: 30 hrs.

Course Code: CCSBS0304	Course Name: OBJECT ORIENTED PROGRAMMING	L	T	P	C
Course Offered in: B. Tech. (CSBS)		2	0	0	2

Pre-requisite: Understand basic programming, data structures, and logic; grasp OOP concepts like classes, objects, and inheritance; and practice debugging

Course Objectives: Students are able to gain a comprehensive understanding of procedural programming in C and object-oriented programming in C++ and Java. They will grasp fundamental OOP concepts, design and develop models using UML tools, and demonstrate standard techniques such as modularity and I/O operations, essential for effective software development.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Identify the concepts of procedural programming and its features.	K2
CO2	Demonstrate the concept of procedural and object-oriented languages.	K3
CO3	Implement fundamental concepts of object-oriented programming using classes and objects.	K3
CO4	Implement reusability and data hiding concepts using C++ and demonstrate generic programming concepts.	K3
CO5	Analyze object-oriented models using UML diagrams.	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO 1	PO2	PO3	PO 4	PO5	PO6	PO 7	PO8	PO9	PO1 0	PO11	PSO1	PSO2	PSO3
CO1	3	2	–	–	2	–	–	–	–	–	1	2	1	1
CO2	3	3	2	1	2	–	–	–	–	–	1	3	2	1
CO3	3	2	2	1	2	–	–	–	–	–	1	3	2	1
CO4	3	3	2	2	2	1	–	–	–	–	2	3	2	2
CO5	3	3	3	2	3	1	–	–	–	–	2	3	3	2

Course Contents / Syllabus

Unit 1	C Foundations & C++ Evolution	8 hours

Procedural programming, An Overview of C: Types Operator and Expressions, Scope and Lifetime, Constants, Pointers, Arrays, and References, Control Flow, Functions and Program Structure, Namespaces, error handling, Input and Output (*C*-way), Library Functions (*string*, *math*, *stdlib*), Command line arguments, Pre-processor directive

Some difference between C and C++: Single line comments, Local variable declaration within function scope, function declaration, function overloading, stronger type checking, Reference variable, parameter passing – value vs reference, passing pointer by value or reference, #define constant vs const, Operator new and delete, the typecasting operator, Inline Functions in contrast to macro, default arguments.

Unit 2	The Fundamentals of OOP and OOP Facilities	7 hours

The Fundamentals of Object-Oriented Programming: Necessity for OOP, Data Hiding, Data Abstraction, Encapsulation, Procedural Abstraction, Class and Object.

More extensions to C in C++ to provide OOP Facilities: Scope of Class and Scope Resolution Operator, Member Function of a Class, private, protected and public Access Specifier, this Keyword, Constructors and Destructors, friend class, error handling (exception)

Unit 3	Essentials of Object-Oriented Programming and Generic Programming	8 hours

Essentials of Object-Oriented Programming: Operator overloading, Inheritance – Single and Multiple, Class Hierarchy, Pointers to Objects, Assignment of an Object to another Object, Polymorphism through dynamic binding, Virtual Functions, Overloading, overriding and hiding, Error Handling

Generic Programming: Template concept, class template, function template, template specialization

Unit 4	Object Oriented Design and Modeling	7 hours
Input and Output: Streams, Files, Library functions, formatted output **Object Oriented Design and Modelling:** UML concept, Use case for requirement capturing, Class diagram, Activity diagram and Sequence Diagram for design, Corresponding C++ code from design		
Total Lecture Hours		**30 hours**

Textbook:

S.No	Book Title	Author
1.	"Object Oriented Modeling and Design", PHI	James Rumbaugh et. Al
2.	Object Oriented Programming with C++,	E Balagurusamy, TMH

Reference Books:

S.No	Book Title	Author
1.	Programming – Principles and Practice Using ++ 2nd Edition	Bjarne Stroustrup, Addison Wesley
2.	The Design and Evolution of C++, 1st Edition	Bjarne Stroustrup, Addison Wesley

Self-Study Content Links:		Duration (in Hrs)
MOOCS	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012972002406719 48837_shared/overview	24 hr 18min
	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013294704351592 44824330_shared/overview	6hr 44 min
Video Lectures	https://www.youtube.com/watch?v=mlIUKyZIUUU	2hr 4 min
	https://www.youtube.com/watch?v=bIzTKJzs92w	4 min 46 sec
	https://www.youtube.com/watch?v=pRC09Tz9iVE	8 min 21 sec
	https://www.youtube.com/watch?v=rr7HVs4d1Qo	16 min 7 sec
Web References	Microsoft Learn – C++ Documentation	
	IBM C++ Documentation	

Mode of Evaluation

CIE							ESE	Total
ST1	ST2	ST3	TA1 5	TA2 5	TA3 5	Attendance 5		
30			20				100	100

LAB Course Code: CCSBS0354	**LAB Course Name: Object Oriented Programming Lab**	**L**	**T**	**P**	**C**
Course Offered in: : B. Tech. (CSBS)		**0**	**0**	**4**	**2**

Pre-requisite: Understand basic programming, data structures, and logic; grasp OOP concepts like classes, objects, and inheritance; and practice debugging

Course Objectives: Students are able to provide students with hands-on experience in implementing object-oriented programming concepts using C++ and Java. This includes designing and developing applications that emphasize key principles such as inheritance, polymorphism, and encapsulation.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Implement classes, objects, inheritance, and polymorphism using C++.	**K3**
CO2	Develop object-oriented applications using UML and OOP principles.	**K3**
CO3	Analyze, debug, and test object-oriented programs for correctness.	**K4**

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**	**PSO3**
CO1	3	2	2	–	3	–	–	–	–	–	1	3	2	1
CO2	2	2	3	2	2	–	–	1	1	1	1	3	3	2
CO3	2	3	2	3	3	–	–	–	1	1	2	2	3	2

List Of Practical's (Indicative & Not Limited To)

Exp. No.	**Experiment Title**	**CO**
1	Implement parameter passing by value, by reference, and passing arrays as constant pointers.	CO1
2	Implement function overloading for string operations (strcat, strncat, strcpy, strncpy).	CO1
3	Implement dynamic memory allocation and deallocation using pointers.	CO3
4	Implement a Complex class with constructor, destructor, copy constructor, and assignment operator using pointer data members.	CO1
5	Implement a Vector class with constructor, destructor, copy constructor, and assignment operator.	CO1
6	Implement a Matrix class with constructor, destructor, copy constructor, and assignment operator.	CO1
7	Implement a Matrix class using Vector with constructor, destructor, copy constructor, and assignment operator.	CO2
8	Implement Stack, Queue, Linked List, Array, and Set classes using access specifiers.	CO1
9	Implement operator overloading for a Complex class (>, =, <=, ==, ++, +, +=, ()).	CO1
10	Implement operator overloading for a Vector class.	CO1
11	Implement operator overloading for a Matrix class.	CO2
12	Implement operator overloading for a Matrix class using Vector.	CO2

13	Implement Stack and Queue classes using inheritance from an Array class.	CO1
14	Implement a template-based Array class with constructor, destructor, copy constructor, assignment operator, and index operator.	CO3
15	Implement template functions for comparison and sorting algorithms (Bubble, Insertion, Merge Sort).	CO3
16	Demonstrate C++ output formatting using manipulators (setw, setprecision, fixed).	CO3
17	Demonstrate C++ input manipulators using getline() and whitespace handling for different data types.	CO3
18	Implement input/output operator overloading (<<, >>) for a Complex class.	CO1
19	Design UML class diagrams and implement Complex, Student, and Book classes.	CO2
20	Design Sequence and Activity Diagrams for a login–logout workflow.	CO2
21	Implement student record management using manipulators and overloaded operators for formatted output.	CO2
22	Implement a custom C++ manipulator for formatting dates (DD-MM-YYYY).	CO3
TOTAL		**60hr**

Course Code: CCSBS0302	Course Name: Computational Statistics	L	T	P	C
Course Offered in: B.Tech (CSBS)		3	0	0	3

Pre-requisite: The student should have basic knowledge of Discrete Mathematics and Statistics & Probability.

Course Objectives:

To develop students' ability to apply computational methods for statistical analysis, inference, and decision-making, while efficiently handling and analysing large datasets using modern computing tools.

Course Outcome: After completion of the course, the student will be able to		**Bloom Level**
CO1	Apply the concept of multivariate normal distribution and its characteristics.	K3
CO2	Apply discriminant analysis techniques used for classification and dimensionality reduction in supervised learning.	K3
CO3	Apply Principal Component Analysis (PCA) in data analysis and machine learning.	K3
CO4	Apply variability among observed and correlated variables in terms of factor analysis	K3
CO5	Apply understanding of clustering techniques in data mining and machine learning.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	-	-	2	-	-	-	-	-	-	2	2	-
CO2	3	3	1	-	2	-	-	-	-	-	-	3	2	-
CO3	3	2	1	-	2	-	-	-	-	-	-	3	2	-
CO4	3	2	-	-	2	-	-	-	-	-	-	3	2	-
CO5	3	2	1	-	2	-	-	-	-	-	-	3	2	-

Course Contents / Syllabus

UNIT 1	**Multivariate Normal Distribution**	**9 hours**

Multivariate Normal Distribution Functions, Conditional Distribution and its relation to regression model, Estimation of parameters

UNIT 2	**Discriminant Analysis**	**9 hours**

Statistical background, linear discriminant function analysis, Estimating linear discriminant functions and their properties.

UNIT 3	**Principal Component Analysis**	**9 hours**

Principal components, Algorithm for conducting principal component analysis, deciding on how many principal components to retain.

UNIT 4	**Factor Analysis**	**9 hours**

Factor analysis model, Extracting common factors, determining number of factors, Transformation of factor analysis solutions, Factor scores.

UNIT 5	**Cluster Analysis**	**9 hours**

Introduction, Types of clustering, Correlations and distances, clustering by partitioning methods, hierarchical clustering, overlapping clustering, K-Means Clustering-Profiling and Interpreting Clusters

Total Lecture Hours	**45 hours**

Textbook:

S.No	Book Title	Author
1.	An Introduction to Multivariate Statistical Analysis,	T.W. Anderson.

Reference Books:

1.	Cluster Analysis for Applications	M.R. Anderberg
2.	Applied Multivariate Data Analysis, Vol I & II.	J.D. Jobson
3.	Statistical Tests for Multivariate Analysis	H. Kris

Self-Study Content Links:		Duration (in Hrs)
MOOCs	Exploratory Multivariate Data Analysis - Course - FUN MOOC	15 hrs
Video Lectures	https://nptel.ac.in/courses/111104024	25 hrs
Web References	Multivariate Data Analysis.pdf	8 hrs

Note: Self-study material may include assignments, seminars, micro projects, industrial visits, any other student activities, etc.

Mode of Evaluation

CIE			**PE**	**Total**
PS1 **10**	**PS2** **20**	**PS3** **20**	**(If mentioned in curriculum)**	
50			**50**	**100**

LAB Course Code: CCSBS0352	LAB Course Name: Computational Statistics Lab	L	T	P	C
Course Offered in B.Tech. CSBS		0	0	2	1

Pre-requisite: Basic Python, Basic Data Structures, and Knowledge of Mathematics Fundamentals

Course Objectives: Apply statistical computing and data science techniques using software tools to analyze datasets, solve real-world business problems, and support data-driven decision-making.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	To Apply Python programming and data wrangling techniques for efficient data processing and analysis	K3
CO2	To analyze Data Aggregation, Group Operations, and Time Series Visualization Techniques Using Python.	K4
CO3	To evaluate the effectiveness of Data Visualization Techniques in Python Including Plotting Graphs, Controlling Graph Properties, Adding Text, and Creating Various Types of Graphs	K5

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	3	1	3	1	-	-	-	-	-	1	2	-
CO2	2	3	3	3	3	1	-	-	-	-	-	1	2	-
CO3	2	2	1	2	1	1	-	-	-	-	-	2	3	-
CO4	3	3	3	3	3	1	-	-	-	-	-	2	3	-

S.No.	Programs	CO & Bloom level
1.	Creation and Manipulation of Vectors, Matrices, Arrays, Lists, Factors, and Data Frames Using Python	CO1, K3
2.	To Create a Python Program for Adding Elements of a Matrix	CO1, K3
3.	Installation of Packages and Execution of Scripts for Importing and Exporting Data Using Python	CO1, K3
4.	To Visualize Statistical Graphs such as Scatter Plots, Box Plots, Histograms, and Pie Charts Using Python	CO3, K5
5.	To Compute Estimators of Statistical Measures such as Mean, Variance, Standard Deviation, Covariance, Correlation, and Standard Error with Suitable Examples and Graphical Representation of Sample Distributions Using Python	CO2, K4
6.	To Write a Python Program for Computing Summary Statistics	CO2, K4
7.	To Plot the Normal Distribution for Class Test Results of a Particular Subject and Identify the Skewness and Kurtosis Using Python	CO2, K4
8.	To Demonstrate Regression Analysis Using Python	CO2, K4
9.	To Perform Clustering of a Dataset and Visualize Clusters Using Principal Component Analysis (PCA) in Python.	CO2, K5

Total Hours: 30 hrs.

Course Code: CCSBS0305	**Course Name:** Software Engineering	**L**	**T**	**P**	**C**
Course Offered in: B.Tech (CSBS)		**2**	**0**	**0**	**2**

Pre-requisite: Basic Knowledge of Design and Analysis of Algorithms

Course Objectives: To enable students to develop methods and procedures for software development that can scale up for large systems and that can be used consistently to produce high-quality software at low cost and with a small cycle of time.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Understand software engineering fundamentals and life cycle models.	K2
CO2	Apply software project planning, estimation, and quality management techniques.	K3
CO3	Analyze software requirements using SRS and UML models.	K4
CO4	Apply object-oriented design principles and software metrics.	K3
CO5	Analyze software testing techniques and quality assurance methods.	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	**PO 1**	**PO2**	**PO3**	**PO 4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO1 0**	**PO11**	**PSO1**	**PSO2**	**PSO3**
CO1	3	2	1	-	-	–	–	–	–	1	-	3	2	2
CO2	3	2	2	2	2	1	–	–	1	2	1	3	3	3
CO3	3	3	2	2	2	–	–	–	1	2	-	3	3	3
CO4	3	3	3	2	3	_	–	–	1	2	-	3	3	3
CO5	3	3	3	3	2	1	–	–	1	2	-	3	3	3

Course Contents / Syllabus

UNIT 1	**Introduction**	**7 hours**

Introduction: Programming in the small vs. programming in the large; software project failures and importance of software quality and timely availability; engineering approach to software development; role of software engineering towards successful execution of large software projects; emergence of software engineering as a discipline.

UNIT 2	**Software Project Management & Software Quality and Reliability**	**8Nhours**

Fundamental Algorithmic Strategies: Basic concepts of life cycle models – different models and milestones; software project planning – identification of activities and resources; concepts of feasibility study; techniques for estimation of schedule and effort; software cost estimation models and concepts of software engineering economics; techniques of software project control and reporting; introduction to measurement of software size; introduction to the concepts of risk and its mitigation; configuration management.

Internal and external qualities; process and product quality; principles to achieve software quality; introduction to different software quality models like McCall, Boehm, FURPS / FURPS+, Dromey, ISO – 9126; introduction to Capability Maturity Models (CMM and CMMI); introduction to software reliability, reliability models and estimation.

UNIT 3	**Software Requirements Analysis, Design and Construction**	**8 hours**

Introduction to Software Requirements Specifications (SRS) and requirement elicitation techniques; techniques for requirement modeling – decision tables, event tables, state transition tables, Petri nets; requirements documentation through use cases; introduction to UML, introduction to software metrics and metrics based control methods; measures of code and design quality.

UNIT 4	**Object Oriented Analysis, Design and Construction & Software Testing**	**7 hours**

Concepts -- the principles of abstraction, modularity, specification, encapsulation and information hiding; concepts of abstract data type; Class Responsibility Collaborator (CRC) model; quality of design; design measurements; concepts of design patterns; Refactoring; object oriented construction principles; object oriented metrics.

Introduction to faults and failures; basic testing concepts; concepts of verification and validation; black box and white box tests; white box test coverage – code coverage, condition coverage, branch coverage; basic concepts of black-box tests – equivalence classes, boundary value tests, usage of state tables; testing use cases; transaction based testing; testing for non-functional requirements – volume, performance and efficiency; concepts of inspection.

Total Lecture Hours		30 hours
Textbook:		
S.No	**Book Title**	**Author**
1.	Software Engineering, , Edition 9, Pearson	Ian Sommerville
Reference Books:		
S.No	**Book Title**	**Author**
1.	Fundamentals of Software Engineering,	Carlo Ghezzi, Jazayeri Mehdi, Mandrioli Dino
2.	Software Requirements and Specification: A Lexicon of Practice, Principles and Prejudices	Michael Jackson
3.	The Unified Development Process	Ivar Jacobson, Grady Booch, James Rumbaugh
4.	Design Patterns: Elements of Object-Oriented Reusable Software	Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides
5.	Software Metrics: A Rigorous and Practical Approach	Norman E Fenton, Shari Lawrence Pfleeger
	Software Engineering: Theory and Practice	Shari Lawrence Pfleeger and Joanne M. Atlee
	Object-Oriented Software Construction, Bertrand Meyer	

Self-Study Content Links:		Duration (in Hrs)
MOOCs	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012605895063871 48827_shared/overview https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012806350984544 25643_shared/overview	27 hr
Video Lectures	https://www.youtube.com/watch?v=NlLM3sVF8wY&t=174s	5hr
Web References	https://www.geeksforgeeks.org/software-engineering/software-engineering/	

Note: Self-study material may include assignments, seminars, micro projects, industrial visits, any other student activities, etc.

Mode of Evaluation

CIE							**ESE**	**Total**
ST1	**ST2**	**ST3**	**TA1 5**	**TA2 5**	**TA3 5**	**Attendance 5**		
30			**20**				**50**	**100**

LAB Course Code: CCSBS0355	LAB Course Name Software Engineering Lab	L	T	P	C
Course Offered in: B. Tech (CSBS)		0	0	4	2

Pre-requisite: Students should have basic knowledge of programming languages such as C/C++ or Java, along with a clear understanding of fundamental data structures like arrays, stacks, queues, linked lists, trees, and graphs.

Course Objectives: Students are able to develop software solutions by applying software engineering principles, integrating various development models and estimation techniques. They can implement quality assurance practices and testing strategies to ensure that the software meets specified standards and requirements

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Apply software solutions using software engineering principles	K3
CO2	Implement and analyze software development models and estimation techniques	K4
CO3	Apply quality assurance practices and testing strategies.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	3	2	2	1	-	-	2	2	1	3	3	2
CO2	3	3	3	3	2	1	-	-	2	2	1	3	3	3
CO3	3	3	3	3	5	1	-	-	2	3	2	3	3	3

List Of Practical's (Indicative & Not Limited To)

Sr. No.	Program	CO, K level
1.	Development of requirements specification on any of the given topic. • Covid vaccination management system • Online grocery store • Online food delivery system • Online medical store • Doctors online OPD	CO1, K3
2.	Develop function-oriented design using SA/SD methodology.	CO2, K3
3.	Develop object-oriented design using UML.	CO3, K3
4.	Designing and implementing test cases manually.	CO3, K3
5.	Designing and implementing test cases manually.	CO3, K3
6.	Use of appropriate CASE tools and other tools (any one) such as configuration management tools, program analysis tools in the software life cycle.	CO3, K3
7.	Create a Software Design Document (SDD): Object and Class diagram.	CO2, K3
8.	Create a Software Design Document (SDD): Object and Class diagram.	CO2, K3
9.	Design test suite for equivalence class partitioning	CO3, K3
10.	Mini Project with CASE tools	CO4, K4

Total Hours: 60 hrs.

Course Code: CNC0303	Course Name : Indian Constitution	L	T	P	C
Course Offered in: B. Tech. (CSBS)		2	0	0	NC

Pre-requisite: Basic understanding of Indian history, civics, the structure of government, and fundamental political concepts.

Students will be able to understand the fundamental principles and features of the Indian Constitution, explain its structure and key provisions, analyze the significance of Fundamental Rights, Fundamental Duties, and the Directive Principles of State Policy, appreciate the roles and responsibilities of the Legislature, Executive, and Judiciary in maintaining democratic governance, and develop an awareness of constitutional values, citizenship, and the importance of upholding the rule of law in everyday life.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Understand the basic features and salient provisions of the Indian Constitution.	K2
CO2	Understand the functioning of the Indian parliamentary system at the Central and State levels.	K2
CO3	Apply the different aspects of the Indian legal system and the functions of its related judicial and constitutional bodies.	K3
CO4	Apply relevant laws, regulations, and ethical principles governing engineering practices in professional scenarios.	K3
CO5	Analyze the role of engineers in relation to government organizations, public institutions, and various governance models.	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	1	1	-	-	-	2	2	1	-	-	1	1	1	1
CO2	1	2	-	-	-	2	1	1	1	1	1	1	1	2
CO3	1	2	-	-	-	3	2	1	-	-	1	2	1	2
CO4	2	2	-	-	-	3	3	2	1	1	1	2	2	3
CO5	2	3	-	-	-	3	2	2	2	2	2	2	2	3

Course Contents / Syllabus

Unit 1	Introduction and Basic Information about Indian Constitution	08 hours

Meaning of the constitution law and constitutionalism, Historical Background of the Constituent Assembly, Government of India Act of 1935 and Indian Independence Act of 1947,Enforcement of the Constitution, Indian Constitution and its Salient Features, The Preamble of the Constitution, Fundamental Rights, Fundamental Duties, Directive Principles of State Policy, Parliamentary System, Federal System, CentreState Relations, Amendment of the Constitutional Powers and Procedure, The historical perspectives of the constitutional amendments in India, Emergency Provisions: National Emergency, President Rule, Financial Emergency, and Local Self Government – Constitutional Scheme in India.

Unit 2	Union Executive and State Executive	08 hours

Powers of Indian Parliament Functions of Rajya Sabha, Functions of Lok Sabha, Powers and Functions of the President, Comparison of powers of Indian President with the United States, Powers and Functions of the Prime Minister, Judiciary – The Independence of the Supreme Court, Appointment of Judges, Judicial Review, Public Interest Litigation, Judicial Activism, Lok Pal, Lok Ayukta, The Lokpa l and Lok ayuktas Act 2013, State Executives – Powers and Functions of the Governor, Powers and Functions of the Chief Minister, Functions of State Cabinet, Functions of State Legislature, Functions of High Court and Subordinate Courts.

Unit 3	Introduction and Basic Information about Legal System	08 hours

The Legal System: Sources of Law and the Court Structure: Enacted law -Acts of Parliament are of primary legislation, Common Law or Case law, Principles taken from decisions of judges constitute binding legal rules. The Court System in India and Foreign Courtiers (District Court, District Consumer Forum, Tribunals, High Courts, Supreme Court). Arbitration: As an alternative to resolving disputes in the normal courts, parties who are in dispute can agree that this will instead be referred to arbitration. Contract law, Tort, Law at workplace.

Unit 4	Intellectual Property Laws and Regulation to Information & Business Organizations and E Governance	06hours

Intellectual Property Laws: Introduction, Legal Aspects of Patents, Filing of Patent Applications, Rights from Patents, Infringement of Patents, Copyright and its Ownership, Infringement of Copyright, Civil Remedies for Infringement, Regulation to Information Introduction, Right to Information Act, 2005, Information Technology Act, 2000, Electronic Governance, Secure Electronic Records and Digital Signatures, Digital Signature Certificates, Cyber Regulations Appellate Tribunal, Offences, Limitations of the Information Technology Act. Sole Traders, Partnerships: Companies: The Company's Act: Introduction, Formation of a Company, Memorandum of Association, Articles of Association, Prospectus, Shares, Directors, General Meetings and Proceedings, Auditor, Winding up. E-Governance and role of engineers in E Governance, Need for reformed engineering serving at the Union and State level, Role of I.T. professionals in Judiciary, Problem of Alienation and Secessionism in few states creating hurdles in Industrial development.

Total Lecture Hours	**30 hours**

Textbook:

S.No	Book Title	Author
1.	Introduction to the Constitution of India	D.D. Basu, LexisNexis, Latest Edition.
2.	Indian Polity	M. Laxmikanth, McGraw Hill Education, Latest Edition
3.	Our Constitution	Subhash C. Kashyap, National Book Trust, India, Latest Edition.

Reference Books:

S.No	Book Title	Author
1.	Constitutional Law of India	J.N. Pandey, Central Law Agency, Latest Edition.
2.	Constitution of India	P.M. Bakshi, Universal Law Publishing, Latest Edition.
3.	The Constitution of India: A Contextual Analysis	Arghya Sengupta, Hart Publishing, Latest Edition

Self-Study Content Links:		Duration (in Hrs)
MOOCs	**NOC: Constitution Law and Public Administration in India, IIT Madras** Prof. Sairam Bhat (12 weeks MOOC Course with more than consolidated 30 (thirty) videos. Link to access course with week-wise videos : nptel.ac.in/courses/129106411 Introductory link of course :https://youtu.be/AqW1m1KRHXg?si=-7K800s7wIKzE_Gi	30
Video Lectures	https://www.youtube.com/live/-9CSdBfgFVg?si=tb4h8VzVhaQZOUM- https://youtu.be/GLeaZrEkL54?si=XpvGXkUTy0Shmfno https://youtu.be/Zn2ivawf3UM?si=E1tHW2nqfZBmsNhe	40 minutes 1 hour
Web References	Constitution of India | My Government | National Portal of India	

Note: Self-study material may include assignments, seminars, micro projects, industrial visits, any other student activities, etc.

Mode of Evaluation

CIE							ESE	Total
ST1	ST2	ST3	TA1 5	TA2 5	TA3 5	Attendance 5		
30			20				50	100

Course Code: CCSBS0403	Course Name: Operating Systems	L	T	P	C
		3	0	0	3

Course Offered in: B. Tech (CSBS)

Pre-requisite: The student should have Basic knowledge of computer fundamentals and organization.

Course Objectives: The objective of this course is to provide an understanding of the basic structure and functions of an operating system and deliver the skills needed to develop and customize Linux shell programs to make effective use of a wide range of standard Linux programming

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Understand the fundamentals, functions, and architecture of Operating Systems.	K2
CO2	Implement and analyze process management policies, CPU scheduling techniques, and thread management in operating systems.	K4
CO3	Understand the need for process synchronization and apply appropriate deadlock handling algorithms to manage resource allocation in concurrent systems.	K3
CO4	Implement and evaluate algorithms for different memory management schemes.	K5
CO5	Analyze and apply fundamental concepts of I/O hardware, file and disk management.	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	1	1	1	-	-	1	1	-	3	3	2
CO2	3	3	3	2	2	1	-	-	2	2	-	3	3	2
CO3	3	3	3	2	2	1	-	2	2	2	-	3	3	2
CO4	3	3	3	2	2	1	-	-	1	1	-	3	3	-
CO5	2	2	2	1	3	1	-	-	1	1	-	3	3	-

Course Contents / Syllabus

Unit 1	**Introduction**	**8 hours**

Concept of Operating Systems (OS), Generations of OS, Types of OS, OS Services, Interrupt handling and System Calls, Basic architectural concepts of an OS, Concept of Virtual Machine, Resource Manager view, process view and hierarchical view of an OS.

Processes: Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching.

Unit 2	**Process Scheduling**	**10 hours**

Thread: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads.

Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time.

Scheduling algorithms: Pre-emptive and non-pre-emptive, FCFS, SJF, RR, Multilevel Queue Scheduling and Multilevel Feedback Queue Scheduling

Unit 3	**Inter- process Communication & Deadlocks**	**10 hours**

Inter-proces communication: Concurrent processes, precedence graphs, Critical Section, Race Conditions, Mutual Exclusion,Hardware Solution, Semaphores, Strict Alternation, Peterson's Solution, The Producer / Consumer Problem, Event Counters, Monitors, Message Passing,Classical IPC Problems: Reader's & Writer Problem, Dinning Philosopher Problem, Barber's shop problem.

Deadlocks: Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.
Concurrent Programming: Critical region, conditional critical region, monitors, concurrent languages, communicating sequential process (CSP); Deadlocks - prevention, avoidance, detection and recovery.

Unit 4	**Memory Management**	**ours**

Memory Management: Basic concept, Logical and Physical address maps, Memory allocation: Contiguous Memory allocation – Fixed and variable partition–Internal and External fragmentation and Compaction, Non Contiguous memory allocation, Paging, Segmentation, Segmentation with paging.

Virtual Memory: Background, Demand paging, Allocation of frames: First Fit, Best Fit, and Worst Fit, Page replacement algorithms (FCFS, Optimal, LRU), Belady's Anomaly, Thrashing

Unit 5	**I/O Hardware, File and Disk Management**	**08 hours**

I**/O Hardware**: I/O devices, Device controllers, Direct Memory Access, Principles of I/O.

File Management: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods (contiguous, linked, indexed), Free-space management (bit vector, linked list, grouping), directory implementation (linear list, hash table), efficiency and performance.

Disk Management: Disk structure, Disk scheduling - FCFS, SSTF, SCAN, C-SCAN, Disk reliability, Disk formatting, Boot-block, Bad blocks.
Case study: UNIX OS file system, shell, filters, shell programming, programming with the standard I/O, UNIX system calls.

Textbook:

Sr. No.	**Book Title**	**Author**
1.	Operating System Concepts Essentials	Abraham Silberschatz

Reference Book:

Sr. No.	**Book Title**	**Author**
1. 2. 3.	Operating System: A Design-oriented Approach, Operating Systems: A Modern Perspective, Design of the Unix Operating Systems,	William Stallings, Edition 9 Charles Patrick Gary J. Nutt. Maurice J. Bach Daniel Pierre Bovet, Marco Cessa
Video Lectures	https://nptel.ac.in/courses/106105214	50 hrs
MOOCS	Complete UNIX & Linux OS Fundamentals Training https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013847808540516352542 74_shared/overview	30hrs 46 mins

Mode of Evaluation

CIE							**ESE**	**Total**
ST1	**ST2**	**ST3**	**TA1** **5**	**TA2** **5**	**TA3** **5**	**Attendance** **5**		
30			**20**				**100**	**150**

LAB Course Code: CCSBS0453	LAB Course Name: Operating Systems Lab (Unix)	L	T	P	C
Course Offered in: B.Tech. (CSBS)		0	0	2	1

Pre-requisite: Proficiency in C programming, Familiarity with control structures, loops, and functions, Knowledge of data structures (queues, linked lists, arrays) for algorithm implementation

Course Objectives: This course gives an ability to students to construct codes for OS API and basics of OS mechanisms and Hands-on and practical experience with usage of the Linux OS and basics of OS Shell Programming

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Implement CPU scheduling, process synchronization, deadlock handling, and memory management techniques.	K3
CO2	Apply memory management, disk scheduling, and file management techniques in operating systems.	K3
CO3	**Apply** commands, shell scripting, and system programming to perform operating system tasks	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2	2	3	-	-	-	-	-	1	3	2	1
CO2	3	3	3	2	3	-	-	-	-	-	1	3	3	1
CO3	2	2	2	2	3	-	-	2	2	1	3	2	3	2

List Of Practical's (Indicative & Not Limited To)

Sr. No.	Program	CO,
1.	Implement FCFS CPU Scheduling algorithm.	CO1
2.	Implement the given CPU Scheduling algorithm a) SJF b) Priority Based	CO1
3.	Implement PRIORITY CPU Scheduling Algorithm (For both Pre-emptive and nonpre-emptive).	CO1
4.	Implement Round-Robin CPU Scheduling Algorithm	CO1
5.	Implement the Producer–consumer problem using semaphores.	CO1
6.	Execute an algorithm for deadlock detection.	CO1
7.	Implement Banker's algorithm of Deadlock Avoidance	CO1
8.	Implement Contiguous memory fixed size partition scheme	CO1
9.	Implement Contiguous memory variable size partition scheme	CO1
10.	Simulate the First-Fit contiguous memory allocation technique.	CO1
11.	Simulate the Best-Fit contiguous memory allocation technique.	CO1
12.	Simulate the Worst-Fit contiguous memory allocation technique	CO1
13.	Implement the Non Continuous Memory Allocation by using Paging.	CO1
14.	Write a Program to simulate the FIFO page replacement algorithm	CO2
15	Write a Program to simulate the LRU page replacement Algorithm	CO2
16	Write a Program to simulate the Optimal page replacement Algorithm.	CO2
17	Write a Program to simulate the FCFS Disk Scheduling Algorithm.	CO2
18	Write a Program to simulate the SSTF Disk Scheduling Algorithm.	CO2

19	Implement SCAN and C-SCAN Disk Scheduling Algorithms.	CO2
20	Implement LOOK and C-LOOK Disk Scheduling Algorithms.	CO2
21	Design an algorithm and implement to organize the file using the single-level directory	CO2
22	Write a program to organize the file using two-level directories.	CO2
23	Write a C program to Sequential files for processing the student information	CO2
24	Write a C program for random access files for processing the employee details	CO2
25	Execute Various types of Linux Commands (Miscellaneous, File oriented, Directory oriented)	CO3
26	Execute a shell program, which accepts the name of a file from standard input and performs the File Readable test on it.	CO3
27	Design and execute the code to accept the name of a file from standard input and performs the File Writable test on it	CO3
28	Implement a shell program, which accepts the name of a file from standard input and performs the File executable test on it.	CO3
29	Implement Linux Networking Commands: ipconfig, traceroute, tracepath, ping, host, hostname, iwconfig.	CO3
30	Implement the following system admin commands in Linux: man, uptime, users, service, pkill, ps	CO3
31	Implement the following in Unix: a) Process creation, b) Sleep Command c) Sleep command using getpid.	CO3
32	Analyse system calls of unix operating system (fork and exit)	CO3
33	Implement Unix commands for a) Signal handling using kil, b) Wait command, c)top	CO3
34	Write a program to simulate UNIX commands like cp, ls, and grep	CO3
35	Implement Unix Shell programming for concatenation of two strings	CO3
36	Implement Unix Shell programming for a) Comparison of two strings b) Maximum of three numbers.	CO3
37	Implement Unix Shell programming for Fibonacci series	CO3
38	Write a program in Unix to whether the given year is a) a leap year or not b) Arithmetic operation using cases.	CO3
39	Write a program in Unix for factorial of a number.	CO3
40	Write a program in Unix to swap the two integers	CO3
41	Write a program in Unix to whether the given number is prime or not.	CO3
	Total Hours: 30 hrs.	

Course Code: CCSBS0401	Course Name: Database Management Systems	L	T	P	C
Course Offered in: B. Tech (CSBS)		3	0	0	3

Pre-requisite: The student should have basic knowledge of discrete mathematics and data structures.

Course Objectives: The objective of the course is to present an introduction to database management systems, with an emphasis on how to organize, maintain and retrieve - efficiently, and effectively - information in different databases.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Apply database uses to solve real world and complex problem and design the ER, EER Model.	K3
CO2	Apply Structured Query Language (SQL) or Procedural Query Language (PL/SQL) to solve the complex queries. Implement relational model, integrity constraints.	K3
CO3	Asses and implement database for storing, managing data efficiently by applying the Normalization process on the database.	K5
CO4	Synthesize the concepts of transaction management, concurrency control and recovery.	K5
CO5	Understand and implement the concepts of Database security and various types of databases.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	2	2	1	1	2	2	2	2	3	2	2
CO2	3	3	2	3	3	1	1	1	2	1	2	3	3	2
CO3	3	2	3	2	3	1	1	2	1	2	2	3	3	2
CO4	2	3	2	3	3	1	2	2	1	2	2	2	3	2
CO5	2	2	2	2	2	3	3	1	2	1	3	3	3	3

Course Contents / Syllabus

Unit 1	**Introduction**	**09 hours**

Introduction to Database. Hierarchical, Network and Relational Models. Database system architecture: Data Abstraction, Data Independence, Data Definition Language (DDL), Data Manipulation Language (DML). Data models: Entity-relationship model, Keys, Mapping constraints, network model, relational and object-oriented data models, integrity constraints, data manipulation operations.

Unit 2	**Relational Query Languages**	**09 hours**

Relational query languages: Relational algebra, Tuple and domain relational calculus, SQL3, DDL and DML constructs, Open source and Commercial DBMS - MYSQL, ORACLE, DB2, SQL server. Relational database design: Domain and data dependency, Armstrong's axioms, Functional Dependencies, Normalization, Normal forms, Normal Forms based on Functional Dependencies (1 NF, 2 NF, 3 NF, BCNF), Multivalued Dependencies (MVDs) and 4NF, Join Dependencies (JDs) and 5NF and Domain Key, Normal Form (DKNF or 6NF), Inclusion Dependencies, Loss-Less Join Decompositions, Dependency preservation, Lossless design, Closure of an attribute set and FD sets, Canonical Cover of FD Sets.

Unit 3	**Query Processing and Optimization**	**09 hours**

Query processing and optimization: Evaluation of relational algebra expressions, Query equivalence, Join strategies, Query optimization algorithms.
Introduction on SQL: Characteristics of SQL, advantage of SQL. SQL data type and literals. Types of SQL commands. SQL operators and their procedure. Tables, views and indexes. Queries and sub queries. Aggregate functions. Insert, update and delete operations, Joins, Unions, Intersection, Minus, Cursors, Triggers, Procedures in SQL/PL SQL.
Storage strategies: Indices, B-trees, Hashing.

Unit 4	**Transaction Processing**	**09 hours**

Concurrency control, ACID property, Serializability of scheduling, Locking and timestamp-based schedulers, Multi-version and optimistic, Concurrency Control schemes, Database recovery. Transaction system, Testing of serializability, serializability of schedules, conflict & view serializable schedule, recoverability, Recovery from transaction failures, log-based recovery, checkpoints, deadlock handling, Locking Techniques for concurrency control, Time stamping protocols for concurrency control, validation-based protocol, multiple granularity, Multi version schemes, Recovery with concurrent transaction, case study of Oracle.		
Unit 5	**Database Security**	**09 hours**
Authentication, Authorization and access control, DAC, MAC and RBAC models, Intrusion detection, SQL injection. Advanced topics: Object oriented and object relational databases, Distributed database Logical databases, Web databases, Distributed databases, Data warehousing and data mining.		
Total Lecture Hours		**45 hours**
Textbook:		
1. 2. 3.	Database System Concepts, Seventh Edition, McGraw – Hill. Fundamentals of Database Systems, Seventh Edition, Addision Wesley. SQL,PL/SQL The programming language Oracle, Fourth Edition, BPB Publication	Korth, Silbertz, Sudarshan Elmasri, Navathe Ivan Bayross
Reference Books:		
1. 2. 3. 4.	Database Systems: A Practical Approach to Design, Implementation and Management, Third Edition, Pearson Education, 2007. Database Management Systems" ThirdEdition, McGrawHill. Implementing Database Security and Auditing, Digital Presss. NoSQL with MongoDB in 24 Hours,First Edition, Sams Publisher	Thomas Cannolly and Carolyn Begg Raghu Ramakrishan and Johannes Gehrke Ron Ben Natan Brad Dayley

Self-Study Content Links:		Duration (in Hrs)
MOOCs	Database Management System Part - 1 - TOC | Infosys Springboard	63 hr 50m
Video Lectures	Normalization Concepts - TOC | Infosys Springboard	1h 15min
Web References	MySQL Documentation	

Mode of Evaluation

CIE							**ESE**	**Total**
ST1	**ST2**	**ST3**	**TA1** **5**	**TA2** **5**	**TA3** **5**	**Attendance** **5**		
30			**20**				**100**	**150**

LAB Course Code: CCSBS0451	LAB Course Name: Database Management Systems Lab	L	T	P	C
Course Offered in: B.Tech (CSBS)		**0**	**0**	**2**	**1**

Pre-requisite: The student should have basic knowledge of discrete mathematics and data structures.

Course Objectives: Students are able to understand the fundamental concepts of database management systems (DBMS) and apply them to design and implement database solutions, including database modeling, querying, and normalization.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Draw and implement the ER and EER model to solve real-world problems. Transform an information model into a relational database schema and utilize data effectively.	K3
CO2	Formulate and evaluate queries using SQL to solve a broad range of query and data update problems.	K5
CO3	Apply and create PL/SQL blocks, procedures, functions, packages, triggers, and cursors. Analyze entity integrity, referential integrity, key constraints, and domain constraints on a database.	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	2	3	1	1	2	2	2	3	3	3	2
CO2	3	3	2	2	3	1	1	1	2	2	3	3	3	2
CO3	3	3	2	2	3	1	2	2	3	3	3	3	3	3

List Of Practical's (Indicative & Not Limited To)

Sr. No.	Program	CO K
1.	Creating Entity-Relationship Diagram using case tools with Identifying (entities, attributes, keys and relationships between entities, cardinalities, generalization, specialization etc.)	CO1 K3
2.	I. Implement DDL commands –Create, Alter, Drop etc. II. Implement DML commands- Insert, Select, Update, Delete	CO1 K3
3.	I. Implement DCL Commands-Grant and Revoke	CO2 K3
4.	II. Implement TCL commands- Rollback, Commit, Save point	CO2 K3
5.	Practice Queries using COUNT, SUM, AVG, MAX, MIN, GROUP BY, HAVING, VIEWS Creation and Dropping.	CO3 K3
6.	Practicing Queries using ANY, ALL, IN, EXISTS, NOT EXISTS, UNION, INTERSECT, CONSTRAINTS etc.	CO3 K3
7.	Practicing Sub queries (Nested, Correlated) and Joins (Inner, Outer and Equi).	CO3 K3
8.	Practicing on Triggers - creation of trigger, Insertion using trigger, Deletion using trigger, Updating using trigger	CO3 K3
9.	Procedures- Creation of Stored Procedures, Execution of Procedure, and Modification of Procedure.	CO2 K3
10.	Cursors- Declaring Cursor, Opening Cursor, Fetching the data, closing the cursor.	CO3 K3
11.	Study of Open Source NOSQL Database: MongoDB (Installation, Basic CRUD operations, Execution) Design and Develop MongoDB Queries using CRUD operations. (Use CRUD operations, SAVE method, logical operators)	CO1 K3

Total Hours: 30 hrs.

Course Code: CCSBS0404	Course Name: Software Design with UML	L	T	P	C
Course Offered in: B.Tech (CSBS)		3	0	0	3

Pre-requisite: Software Engineering concepts.

Course Objectives: Students will understand the importance of modeling in the software development life cycle. They can apply the object-oriented approach to analyze and design systems and software solutions. They will understand how to employ the UML notation to create effective and efficient system designs.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Understand the object-oriented approach to analysing and designing systems and software solutions.	K2
CO2	Understand and become familiar with the Unified modelling Language	K2
CO3	Analyze the requirements through use case driven approach.	K4
CO4	Demonstrate the logical view of system using class diagram model.	K3
CO5	Analyze the conceptual model into various scenarios and applications.	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	-	2	-	-	-	1	-	2	3	2	2
CO2	2	-	2	-	2	-	-	-	1	-	1	2	3	3
CO3	2	3	2	-	1	-	-	-	1	-	1	2	2	2
CO4	3	2	3	2	2	-	-	-	1	-	1	2	2	2
CO5	3	2	3	2	2	1		1	2	1	2	3	3	3

Course Contents / Syllabus

UNIT 1	**Software development process**	**09 hours**

The Waterfall Model vs. The Spiral Model, The Software Crisis, description of the real world using the Objects Model, Classes, inheritance and multiple configurations, Quality software characteristics, Description of the Object-Oriented Analysis process vs. the Structure, Study of approaches Cord &Yordon, Graddy Booch, James Raumbaugh.

UNIT 2	**Introduction to the UML Language**	**09 hours**

Standards, Elements of the language, General description of various models, The process of Object-Oriented software development, Design Patterns, and its types.

UNIT 3	**Requirement s Analysis Using Case Modeling**	**09 hours**

Analysis of system requirements, Actor definitions. Writing a case goal, Use Case Diagrams, Use Case Relationships Interaction Diagrams: Description of goal, Defining UML Method, Operation, Object Interface, Class, Sequence Diagram, Collaboration Diagram.

UNIT 4	**The Logical View Design Stage**	**09 hours**

Diagrams: The Class Diagram Model, Attributes descriptions, Operations descriptions, Connections descriptions in the Static Model, Association, Generalization, Aggregation, Dependency, Interfacing, and Multiplicity. Package Diagram Model: Description of the model: White box, black box, Connections between packagers. Interfaces. Create a Package Diagram.

UNIT 5	**Models**	**09 hours**

Dynamic Model: State Diagram / Activity Diagram, Description of the State Diagram, Events Handling, Description of the Activity Diagram, Exercise in State Machines.Component Diagram Model: Physical Aspect. Logical Aspect, Connections and Dependencies, User face.Deployment Model: Processors, Connections, Components, Tasks, Threads, Signals and Events.

Total Lecture Hours	**45 hours**

Textbook:

Sr. No.	**Book Title**	**Author**

1. 2.	applying-uml-and-patterns- https://personal.utdallas.edu/~chung/SP/applying-uml-and-patterns.pdf Blueprints: Creating, Describing, and Implementing Designs for Larger-scale Software Projects: https://stephendavies.org/blue.pdf	Craig Larman. Stephen Davies.

Reference Books:

Sr. No.	Book Title	Author
1.	Design Patterns: Elements of Reusable Object-Oriented Software	Erich Gamma, Richard Helm, Ralph Johnson, and John M.
2.	Object-Oriented Systems Analysis and Design Using UML (4th Edition).	Simon Bennett, Steve McRobb, Ray Farmer

Self-Study Content Links:		Duration (in Hrs)
MOOCs	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013501592357027840012540/overview https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013841706682785792183 3_shared/overview https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013501576262418432102 34/overview https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_014157698265661440226/overview	1 hr 16m 6 hr 10m 1 hr 8m 10h 31m
Video Lectures	https://www.bilibili.com/video/BV1uY41137fS	20h
	https://youtu.be/tZreaH_FyMs?si=bXTDr3lATem_0h-M	18h
Web References	https://www.omg.org/uml	
	https://www.omg.org/uml/	

Mode of Evaluation

CIE							ESE	Total
ST1	ST2	ST3	TA1 5	TA2 5	TA3 5	Attendance 5		
30			20				100	150

LAB Course Code: CCSBS0454	LAB Course Name: Software Design with UML Lab	L	T	P	C
Course Offered in: B. Tech (CSBS)		**0**	**0**	**2**	**1**

Pre-requisite: Software Engineering concepts

Course Objectives: Students are able to learn and apply UML modeling techniques for software design, including conceptual model

Course Outcome: After completion of the course, the student will be able to		**Blooms Level**
CO1	Identify and Understand ambiguities, inconsistencies, and incompleteness in a requirements specification, and articulate functional and non-functional requirements.	K2
CO2	Identify classes and their associations, and draw class diagrams. Graphically represent various UML diagrams, including associations, and depict the logical sequence of activities in a system.	K3
CO3	Apply and Analyze UML modeling techniques to design and develop software systems. Emphasize conceptual modeling, structural and behavioral diagrams, and system documentation.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	3	2	2	-	-	-	-	2	-	2	3	2	2
CO2	2	2	3	-	3	-	-	-	2	-	2	2	3	2
CO3	3	2	3	3	3	-	-	-	3	1	3	3	3	3

List Of Practical's (Indicative & Not Limited To)

1. Use Case Diagram.	CO1
2. Use Case Diagram for ATM.	CO1
3. Use Case Diagram for Stock Maintenance System	CO1
4. Use Case Diagram for Remote Procedure Call.	CO1
5. Class Diagram.	CO1
6. Class Diagram for ATM.	CO1
7. Class Diagram for Stock Maintenance System.	CO1
8. Object Diagram.	CO1
9. Object Diagram for ATM.	CO1
10. Object Diagram for Stock Maintenance System.	CO1
11. Sequence Diagram.	CO1
12. Sequence Diagram for ATM.	CO2
13. Sequence Diagram for Stock Maintenance System.	CO2
14. Collaboration Diagram.	CO2
15. Collaboration Diagram for ATM.	CO2
16. Collaboration Diagram for Stock Maintenance System.	CO2

17. Collaboration Diagram for Remote Procedure Call.	CO2
18. State Chart Diagram.	CO2
19. State Chart Diagram for ATM.	CO3
20. State Chart Diagram for Stock Maintenance System.	CO3
21. Activity Diagram.	CO3
22. Activity Diagram for ATM.	CO3
23. Activity Diagram for Stock Maintenance System.	CO3
24. Component Diagram.	CO3
25. Component Diagram for ATM.	CO3
26. Deployment Diagram.	CO3
Total 30 hrs	

Course Code: CCSBS0406	Course Name: Introduction to Innovation, IP Management & Entrepreneurship	L	T	P	C
Course Offered in: B.Tech (CSBS)		**3**	**0**	**0**	**3**

Pre-requisite: Good knowledge of Fundamentals of Innovation in Entrepreneurship.

Course Objectives: This course is intended to inculcate the knowledge and application of innovation in business processes. This course would also make the students capable of identifying the opportunities and setting up entrepreneurial venture complying with prevailing intellectual property rights.

Course Outcome: After completion of the course, the student will be able to

Course Outcome: Student will be able to :		Bloom's Knowledge Level (KL)
CO1	Understand the concept and importance of innovation in business.	K2
CO2	Apply the concepts of innovation in real world issues in order to create new ventures	K3
CO3	Understand the entrepreneurial opportunities in order to secure competitive advantage of business.	K2
CO4	To analyze the available funding sources for financing the projects	K4
CO5	To understand and apply the knowledge of IPRs in business	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	2	1	1	2	2	2	1	2	1	2	2	2
CO2	3	3	3	2	2	2	3	2	2	2	2	3	3	3
CO3	2	2	3	2	2	3	3	2	2	2	2	3	3	3
CO4	2	3	2	3	2	2	2	2	2	2	3	2	2	2
CO5	2	2	2	2	2	3	2	3	2	2	2	2	2	3

Course Contents / Syllabus

Unit 1	**Innovation**	**08 hours**

Innovation: What and why? Innovation as a core business process, Sources of innovation, Knowledge push vs. need pull innovations ,Types of Innovation

Unit 2	**Building an Innovative Organization**	**09 hours**

Creating new products and services, Go- it- alone approach Exploiting open innovation and collaboration, Use of innovation for starting a new venture

Unit 3	**Entrepreneurship**	**08 hours**

Opportunity recognition and entry strategies, Entrepreneurship as a Style of Management, Maintaining Competitive Advantage- Use of IPR to protect Innovation.

Unit 4	**Financial Planning**	**10 hours**

Financial Projections and Valuation, Stages of financing, Debt, Venture Capital and other forms of Financing.

Unit 5	**Intellectual Property Rights (IPR)**	**10 hours**

Genesis and Development, International Context, Concept of IP Management, Use in marketing. Types of Intellectual Property Patent- Procedure, Licensing and Assignment, Infringement and Penalty, Trademark- Use in marketing, example of trademarks- Domain name, Geographical Indications- What is GI, why protect them? Copyright- What is copyright? Industrial Designs- What is design? How to protect?

Total Lecture Hours	**45 hours**

Textbook:

Sr. No.	Book Title	Author

1. Joe Tidd, John Bessant. Managing Innovation: Integrating Technological, Market and Organizational Change, joe Tidd, John Bessant.		
Reference Books:		
Sr. No.	**Book Title**	**Author**
1. Managing Innovation and Entrepreneurship, Robert D. Hisrich, Claudine Kearney		

Self-Study Content Links:		Duration (in Hrs)
MOOCs	Entrepreneurship: Launching an Innovative Business | Coursera	40 Hrs
Web References	Entrepreneurship Course - TOC | Infosys Springboard	6 Hrs

Mode of Evaluation

CIE							**ESE**	**Total**
ST1	**ST2**	**ST3**	**TA1**	**TA2**	**TA3**	**Attendance**		
			5	**5**	**5**	**5**		
30			**20**				**100**	**150**

Course Code: CCSBS0402	Course Name: Operations Research	L	T	P	C
Course Offered in: B. Tech (CSBS)		2	0	0	2

Pre-requisite: Basic Knowledge of Engineering mathematics along with logical and analytical problem solving skills.

Course Objectives: After completion of the course, the student will be able to apply the concepts of Operations research.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Apply suitable decision-making methods and Linear Programming Problems (LPP) techniques to solve real-world business and industry problems effectively.	K3
CO2	Apply Transportation problems to optimize real-life distribution and logistics	K3
CO3	Apply Assignment techniques for effective real-world decision-making.	K3
CO4	Apply Simulation techniques for effective real-world decision-making.	K3
CO5	Apply CPM, PERT, and Inventory Control to reduce cost and time in production and manufacturing.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	2	-	-	-	-	-	-	-	2	2	3
CO2	3	3	1	3	-	-	-	-	-	-	1	3	2	3
CO3	3	2	1	3	-	-	-	-	-	-	2	3	2	3
CO4	3	2	1	3	-	-	-	-	-	-	2	3	2	3
CO5	3	3	1	3	-	-	-	-	-	-	1	3	2	3

Course Contents / Syllabus

Module 1	**Introduction to OR and Linear Programming Problem**	**9 hours**

Introduction to OR, Formulation of different OR models , definitions of Hyper plane, Convex set, Concave set.

Linear programming -Linear Programming Problem (LPP), Generalized LPP- Formulation of L.P.P, Geometric method or Graphical method of LPP, Simplex Method , Big-M method, Two phase method, Duality and Dual Simplex Method.

Module 2	**Transportation Problem**	**7 hours**

Transportation Problem: -Formulation of Transportation problem, Balanced & Unbalanced problems, NWCR method, LCM method and VAM method, test for optimality (MODI method), degeneracy and its resolution.

Module 3	**Assignment Problem & Simulation Methodology**	**7 hours**

Assignment Problem: - Formulation of Assignment problem, Initial B.F.S. of Balanced & unbalanced Assignment problems, Hungarian Method.

Simulation Methodology: Definition and steps of simulation, Random number, Random number generator, Discrete Event System Simulation – clock, event list, Monte-Carlo simulation, Applications of simulation.

Module 4	**PERT-CPM & Inventory Control**	**7 hours**

Network models & Project Management : Construction of Network**,** Forward Pass and Backward Pass Calculation, Critical Path Method (CPM) , Program Evaluation and Review Technique (PERT) , Time Estimates in PERT, Difference between PERT & CPM.

Inventory Control: Functions of inventory and its disadvantages, Concept of inventory costs, Basics of inventory policy (order, lead time, types), Fixed order-quantity models – EOQ (only Deterministic model)

Total Lecture Hours	**30 hours**

Textbook:

Sr. No.	Book Title	Author
1.	Operations Research: An Introduction	Taha, H. A.
2.	Operations Research	S. kalavathy
3.	Operations Research Theory and Applications	J.K.Sharma

Reference Books:		
Sr. No.	**Book Title**	**Author**
1.	Operations Research	Kanti Swarup, P. K. Gupta & Man Mohan
2.	Quantitative Analysis for Management	Render, Stair & Hanna –

Moocs: Operations Research https://www.acsce.edu.in/acsce/wp-content/uploads/2020/03/1585041316993_Module-4.pdf	32 Hrs
Nptel Link: https://nptel.ac.in/courses/126105663	10 hrs

Mode of Evaluation

CIE							**ESE**	**Total**
ST1	**ST2**	**ST3**	**TA1**	**TA2**	**TA3**	**Attendance**		
			5	**5**	**5**	**5**		
30			**20**				**50**	**100**

LAB Course Code: CCSBS0452	LAB Course Name: Operations Research Lab	L	T	P	C
Course Offered in: B.Tech.		**0**	**0**	**2**	1
Pre-requisite: Knowledge of Mathematics Fundamentals					
Course Objectives: To develop the ability to use statistical software and data science techniques for analyzing complex datasets, generating insights, and addressing real-world problems.					
Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)			
CO1	Apply Python programming to solve Linear Programming Problems using graphical, simplex, Two-Phase, Big M, and Dual Simplex methods.	K3			
CO2	Analyze optimization problems by implementing Transportation and Assignment algorithms (NWCR, VAM, and Hungarian Method using Python)	K4			
CO3	Evaluate computational results obtained from operation research algorithm to support real-world problems of business and industry using python.	K5			

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	**3**	**2**	**1**	**2**	-	-	-	-	-	-	-	**2**	2	2
CO2	**3**	**3**	**1**	**3**	-	-	-	-	-	-	-	**2**	2	2
CO3	**3**	**2**	**1**	**3**	-	-	-	-	-	-	-	**3**	1	1

S.NO.	Programs	CO and Bloom level
1.	To Solve a Linear Programming Problem Using the Graphical Method in Python.	CO1,K3
2.	To Enumerate Basic Feasible Solutions of a Linear Programming Problem Using Python.	CO1,K3
3.	To Solve a Linear Programming Problem Using the Simplex Method in Python.	CO1,K3
4.	To Solve a Linear Programming Problem Using the Big M Method in Python.	CO1,K3
5.	To Solve a Linear Programming Problem Using the Two-Phase Method in Python.	CO1,K3
6.	To Solve a Linear Programming Problem Using the Dual Simplex Method in Python.	CO1,K3
7.	To Solve a Transportation Problem Using North west corner method (NWCR) in Python.	CO2,CO3, K5
8.	To Solve a Transportation Problem Using Vogel's Approximation Method (VAM) in Python.	CO2,CO3, K5
9.	To Solve an Assignment Problem Using the Hungarian Method in Python.	CO2,CO3, K5

Total Hours: 30 hrs.

Course Code: CCSBS0405	Course Name: Design Thinking	L	T	P	C
Course Offered in: B. Tech. CSBS		2	0	0	2

Pre-requisite: Basic knowledge of business fundamentals, problem-solving, and effective communication skills.

Course Objectives: The objective of this course is to familiarize students with design thinking process as a tool for breakthrough innovation. It aims to equip students with design thinking skills and ignite the minds to create innovative ideas, develop solutions for real-time problems.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Develop a strong understanding of the design process and apply it in a variety of business settings	K3
CO2	Understand the user, empathize and implement it to real life scenario	K3
CO3	Formulate specific problem statements of real time issues and generate innovative ideas using design tools	K5
CO4	Apply critical thinking skills to arrive at the root cause from a set of likely causes	K4
CO5	Demonstrate an enhanced ability to apply design thinking skills for evaluation of claims and arguments	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	1	2	3	1	2	2	1	2	2	3	2	3	2	2
CO2	1	2	2	1	1	3	2	3	3	1	2	2	2	3
CO3	2	3	3	2	3	2	2	2	2	2	2	3	3	2
CO4	2	3	2	3	2	1	1	1	2	1	2	3	2	2
CO5	2	3	3	3	2	2	2	2	3	2	3	3	2	3

Course Contents / Syllabus

Unit 1	Introduction	8 hours

Introduction to design thinking, traditional problem solving versus design thinking, history of design thinking, wicked problems. Innovation and creativity, the role of innovation and creativity in organizations, design mindset, Examples of good, bad and Great Design, Introduction to elements and principles of design, 13 Musical Notes for Design Mindset, Design Approaches across the world

Case Studies: Mumbai Dabbawallas, Gillette, Singapore, Bengaluru, Bahubali, Google, Embrace Incubator

Activity: Observation, Wicked Problem

Unit 2	Empathy	8 hours
Understanding stakeholders, techniques to empathize with, identify key user problems. Empathy tools- Interviews, empathy maps, persona, immersion and observations, customer journey maps, Classifying insights after Observations, Classifying Stakeholders. **Case Studies:** Pure-it, Royal Enfield, Big Basket, Air-bnb. **Activity:** Moccasin Walk, Persona, Empathy map, Journey Map		
Unit 3	**Define & Ideation**	**6 hours**
Defining the problem statement, Finding Root cause of the problem: 5 Why's, 4W's, Point of View (POV) statements, How might We and Defining the problem using Ice-Cream Sticks, Market Research and it's types Idea Generation-basic design directions, Themes of Thinking, brainstorming, Do's & Don'ts for Brainstorming, Increase the association, random association technique, Metaphor, ideation activity games – Mindmap, Six Thinking Hats, Million Dollar idea **Case Studies:** The Good Kitchen, Flipkart, Uber, Redbus **Activity:** 5 Why, HMW, Brainstorming, Six Thinking Hats		
Unit 4	**Prototyping & Testing**	**8 hours**
Prototyping (Convergence): Prototyping mindset, tools for prototyping – Sketching, paper models, pseudo-codes, physical mockups, Interaction flows, storyboards, acting/role-playingetc, importance of garnering user feedback for revisiting Brainstormed ideas, Idea Selection: Refine and narrow down to the best idea, 10-100-1000gm, QBL, Design Tools for Convergence – SWOT Analysis for 1000gm discussion, Minimum Viable Prototype. **Storytelling: elements of storytelling, mapping persona with storytelling, elevator pitch** Napkin Pitch**, and inspirations and references some examples of Successful campaign** A/B Testing, Decision Making Tools and Approaches – Vroom Yetton Matrix, Shift-Left, Up, Right, Value Proposition, Testing of design with people, conducting usability test, testing as hypothesis, testing as empathy, observation and shadowing methods, Guerrilla Interviews, validation workshops, user feedback, record results, enhance, retest, and refine design, Software validation tools, design parameters, alpha & beta testing, Taguchi, defect classification, random sampling. Agile Methodology. Satori **Case Studies:** Big Bazaar **Activity:** 30 Circles, paper prototype, roleplay, Ad-Mad Show		
Total Lecture Hours		**30 hours**

Textbook:

S.No	Book Title with publication agency & year	Author
1.	UnMukt : Science & Art of Design Thinking	Arun Jain
2.	Solving Problems with Design Thinking – Ten Stories of What Works	Jeanne Liedta

Reference Books:

S.No	Book Title with publication agency & year	Author

1. Vijay Kumar, 101 Design Methods: A Structured Approach for Driving Innovation in Your Organization, 2013, John Wiley and Sons Inc, New Jersey
2. BP Banerjee, Foundations of Ethics and Management, 2005, Excel Books
3. Gavin Ambrose and Paul Harris, Basics Design 08: Design Thinking, 2010, AVA Publishing SA
4. Roger L. Martin, Design of Business: Why Design Thinking is the Next Competitive Advantage, 2009, Harvard Business Press, Boston MA

Self-Study Content Links:		Duration (in Hrs)
MOOCs	1- Design Thinking & Project Management: 2 in 1 Complete Guides https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_014157697564917760227/overview 2- Designing the Future of Work https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_014235404506750976759/overview	28 Hrs 04-06 Hrs
Video Lectures	https://youtu.be/rUUuhnLkJ2s?si=_XCHnDbt_U1z0Frx https://www.youtube.com/watch?v=ldYzbV0NDp8 https://www.youtube.com/watch?v=0Fi83BHQsMA	06 min 02 min 14 min
	https://www.youtube.com/watch?v=q654-kmF3Pc https://swayam.gov.in/nd1_noc19_mg60/preview	04 min 08 weeks
	https://www.udemy.com/course/design-thinking-for-beginners/ https://www.interaction-design.org/literature/article/personas-why-and-how-you-should-use-them	1 hr
	https://www.youtube.com/watch?v=GsF4reOqsGE	11 min
Web References	NA	

Mode of Evaluation

CIE							**ESE**	**Total**
ST1	**ST2**	**ST3**	**TA1** **5**	**TA2** **5**	**TA3** **5**	**Attendance** **5**		
30			**20**				50	**100**

LAB Course Code: CCSBS0455	LAB Course Name: Design Thinking Lab	L	T	P	C
Course Offered in: B.Tech CSBS		0	0	2	1

Pre-requisite: Basic knowledge of business fundamentals, problem-solving, and effective communication skills.

Course Objectives:

The objective of this course is to familiarize students with design thinking process as a tool for breakthrough innovation. It aims to equip students with design thinking skills and ignite the minds to create innovative ideas, develop solutions for real-time problems.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO 1	Develop a strong understanding of the design process and apply it in a variety of business settings	K3
CO 2	Understand the user, empathize and implement it to real life scenario	K3
CO 3	Formulate specific problem statements of real time issues and generate innovative ideas using design tools	K5

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO 2	PSO2
CO1	1	2	2	1	1	2	1	2	2	2	2	3	2	2
CO2	1	2	1	1	1	1	2	2	2	1	2	2	3	2
CO3	1	2	1	2	2	2	1	2	2	1	2	3	3	3

List Of Practical's (Indicative & Not Limited To)

Sr. No.	Program	CO
1.	Moccasin Walk activity for 1 hour to allow learners experience stepping into the shoes of another person. This is an individual activity. Sharing observations with the group. Suggest that students try this even in their free time away from studies. (Imagine designing a solution for people who struggle with grocery shopping. A Moccasin Walk might involve accompanying a user on their regular grocery trip, observing their navigation through the store, noting what products they choose, and how they interact with the checkout process. This firsthand experience can reveal challenges and opportunities that might not be apparent from a distance.)	CO2
2.	**Immersion activity: Immersion activity** Participants will be divided into four groups. Each group will need to visit any one of the following places to conduct an immersion activity. They need to interview people and fill up the DT question template (explained in the last class) Accounts Office, Admission / Placement Department, College cafeteria, College library, College sports facility, Transport facility near college	CO2

3.	**Defining problem statements** : Group activity, in which each group will define the key problem statements (max three) for their lead personas. Each group will present while the remaining groups will do a peer review. Finally, lecturer will moderate/validate the problem statements (based on handouts provided by DT Team)	CO3
4.	**Ideation games Game** 1: Six Thinking Hats, Game 2: Million-dollar idea	CO3
5.	**Ideate to find solutions**: Participants will work in their assigned groups to ideate solutions for the problem statements they identified (as continuation of immersion activity) applying ideation methods discussed in the previous session. They will get scores based on how well they can apply the ideation methods.	CO3
6.	**Prototype your idea:** This is a group activity in which the participants will work in groups (created at the beginning of the course, in which they did immersion, persona creation, defining problem statement and ideating) to create prototypes based on the solutions they had identified.	CO1
7.	**Test the Prototype:** Each group needs to test their prototype created earlier and: 1. Document user feedback 2. Write down their inference from the feedback 3. Give test cases	CO1
8.	**8,9,10: Project** **Option 1:** Each group needs to present a Prototype of how they can apply DT in their functional work or coding. Examples will be provided to explain what exactly they need to do. **Option 2:** Each group will apply DT to create a prototype to improve any existing product or service. For both options, groups need to complete all phases of the Stanford DT model and include the outputs of each phase in their presentation. Lecturers will evaluate the project based on the rubric provided by the DT Team.	CO1

Total Hours: 30hrs.

Course Code: CNC0404	Course Name: Essence of Indian Traditional Knowledge	L	T	P	C
Course Offered in: B. Tech. CSBS		2	0	0	2

Pre-requisite: A basic interest in Indian culture, heritage, values, and philosophical thought will help learners better understand and appreciate the concepts discussed in the course.

Course Objectives: The objective of this course is to introduce learners to the essence of Indian tradition and knowledge systems by fostering an understanding of India's rich cultural, philosophical, scientific, and ethical heritage. The course aims to develop awareness of core Indian values such as harmony, sustainability, duty, and holistic living, and to enable students to apply this timeless wisdom in personal, social, and professional life for balanced individual growth and societal well-being.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Understand the basics of past Indian politics and state polity	K2
CO2	Understand the Vedas, Upanishads, languages & literature of Indian society.	K2
CO3	Know the different religions and religious movements in India.	K2
CO4	Identify and explore the basic knowledge about the ancient history of Indian agriculture, science & technology, and ayurveda.	K2
CO5	Correlate role of engineers with different organizations and governance models	K2

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	-	-	-	-	-	-	-	-	2	1	-	-
CO2	-	-	-	-	-	-	3	2	-	-	2	-	-	-
CO3	-	2	-	-	-	-	-	3	-	-	-	-	-	2
CO4	1	-	-	-	-	2	-	3	-	-	2	-	-	-
CO5	1	-	-	-	3	-	-	-	-	2	-	-	-	2

Course Contents / Syllabus

Unit-1	Society State and Polity in India	07 hours

Evolutionary Theory, Force Theory, Mystical Theory Contract Theory, Stages of State Formation in Ancient India, Kingship , Council of Ministers Administration Political Ideals in Ancient India Conditions' of the Welfare of Societies.

The Seven Limbs of the State, Society in Ancient India, Purusārtha, Varnāshrama System, Āshrama or the Stages of Life, Marriage, Understanding Gender as a social category, The representation of Women in Historical traditions, Challenges faced by Women. Four-class Classification, Slavery.

Unit-2	Indian Literature, Culture, Tradition, and Practices	08 hours

Evolution of script and languages in India: Harappan Script and Brahmi Script. The Vedas, the Upanishads, the Ramayana and the Mahabharata, Puranas, Buddhist And Jain Literature in Pali, Prakrit And Sanskrit, Kautilya's Arthashastra. Famous Sanskrit Authors, Telugu Literature, Kannada Literature, Malayalam Literature, Sangama Literature Northern Indian Languages & Literature, Persian And Urdu ,Hindi Literature

Unit-3	Indian Religion, Philosophy, and Practices	08 hours

Pre-Vedic and Vedic Religion, Buddhism, Jainism, Six System Indian Philosophy, Shankaracharya, Various Philosophical Doctrines, Other Heterodox Sects. Bhakti Movement, Sufi movement, Socio religious reform movement of 19th century, Modern religious practices.

Unit-4	Science, Management, Indian Knowledge System, Cultural Heritage and Performing Arts	07 hours

Astronomy in India, Chemistry in India, Mathematics in India, Physics in India, Agriculture in India, Medicine in India, Metallurgy in India, Geography, Biology Harappan Technologies, Water Management in India, Textile Technology in India, Writing Technology in India Pyrotechnics in India Trade in Ancient India/India's Dominance up to Precolonial Times. Indian Architect, Engineering and Architecture in Ancient India, Sculptures, Seals, coins, Pottery, Puppetry, Dance, Music, Theatre,

drama, Painting, Martial Arts Traditions, Fairs and Festivals, Current developments in Arts and Cultural, Indian's Cultural Contribution to the World. Indian Cinema.	
Total Lecture Hours	**30 hours**
Textbooks:	
	Authors Name
Designed as a systematic introduction to Indian knowledge heritage, aligned with NEP 2020 goals to integrate traditional wisdom with modern education.	*Jaikishan Sahu & Pavan Kumar Mittal*
Reference Books:	
	Authors Name
Traditional Knowledge System and Technology in India A comprehensive textbook-style survey of continuous Indian intellectual traditions across philosophy, epistemology, and cultural thought	Basanta Kumar *Kapil Kapoor & Avadhesh Kumar Singh*

Self-Study Content Links:		**Duration (in Hrs)**
MOOCs	Mastering VMware vSphere 6.5	16Hrs 08Min
Video Lectures	https:/ https://youtu.be/VVpq6mO_WGk?si=VIx6AC0uwnASQKb6 /youtu.be/yXiSNNfyBeE?si=tmTwT3Dwvz5vfG_V	00Hrs 25Min 01Hrs 51Min 03Hrs 48Min 10Hrs 17Min
Web References	https://youtu.be/_YzG4FRn7GA?si=i2Z055SXQaa_CRex https://www.youtube.com/live/gRHgFC5iwWA?si=InVg2T8f1ObuO4Un	

Mode of Evaluation

CIE							**ESE**	**Total**
ST1	**ST2**	**ST3**	**TA1** **5**	**TA2** **5**	**TA3** **5**	**Attendance** **5**		
30			**20**				50	100