NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY, GREATER NOIDA, GAUTAM BUDDH NAGAR
(AN AUTONOMOUS INSTITUTE)

Affiliated to

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

Evaluation Scheme & Syllabus
For

Bachelor of Technology
Computer Science and Business Systems

Second Year

(Effective from the Session: 2025-26)

NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY, GREATER NOIDA, GAUTAM BUDDH NAGAR
(AN AUTONOMOUS INSTITUTE)

Bachelor of Technology
Computer Science and Business Systems

Evaluation Scheme

SEMESTER-III

Sl. No.	Subject Codes	Subject	Types of Subjects	Periods		Evaluation Schemes					End Semester		Total	Credit
				L	T	P	CT	TA	TOTAL	PS	TE	PE		
1	BCSBS0306Z	Formal Language and Automata Theory	Mandatory	3	1	0	30	20	50		100		150	4
2	BCSBS0303	Computer Organization & Architecture	Mandatory	3	0	0	30	20	50		100		150	3
3	BCSBS0301	Computational Statistic	Mandatory	3	0	0	30	20	50		100		150	3
4	BCSBS0302Z	Object Oriented Programming	Mandatory	2	0	0	30	20	50		50		100	2
5	BCSBS0304Z	Software Engineering	Mandatory	2	0	0	30	20	50		50		100	2
6	BCSBS0353	Computer Organization & Architecture Lab	Mandatory	0	0	2				25		25	50	1
7	BCSBS0352	Object Oriented Programming Lab	Mandatory	0	0	4				50		50	100	2
8	BCSBS0351	Computational Statistic Lab	Mandatory	0	0	2				25		25	50	1
9	BCSBS0354Z	Software Engineering Lab	Mandatory	0	0	4				50		50	100	2
10	BNC0303	Indian Constitution	Compulsory Audit	2	0	0	30	20	50			50	100	
		Massive Open Online Courses (For B.Tech. Hons. Degree)	MOOCs											
		TOTAL		**15**	1	**12**	**180**	**120**	**300**	**150**	**400**	**200**	**950**	**20**

*** List of MOOCs Based Recommended Courses for Second year (Semester-III) B. Tech Students**

Sr. No.	Subject Code	Course Name	University / Industry Partner Name	No of Hours	Credits
1	BMC0052	Computational Problem Solving	Infosys Wingspan (Infosys Springboard)	27h 26m	2
2	BMC0051	Software Defined Networking	Infosys Wingspan (Infosys Springboard)	32h 28m	2.5
3	BMC0053	TechA Linux Programming Foundation Certification	Infosys Wingspan (Infosys Springboard)	19h	1.5

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 - BNC0303)**

➢ 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, GAUTAM BUDDH NAGAR
(AN AUTONOMOUS INSTITUTE)
Bachelor of Technology
Computer Science and Business Systems

Evaluation Scheme
SEMESTER-IV

Sl. No.	Subject Codes	Subject	Types of Subjects	Periods			Evaluation Schemes				End Semester		Total	Credit
				L	T	P	CT	TA	TOTAL	PS	TE	PE		
1	BCSBS0403	Operating Systems	Mandatory	3	0	0	30	20	50		100		150	3
2	BCSBS0404	Database Management Systems	Mandatory	3	0	0	30	20	50		100		150	3
3	BCSBS0402Z	Software Design with UML	Mandatory	3	0	0	30	20	50		100		150	3
4	BCSBS0405Z	Introduction to Innovation, IP Management & Entrepreneurship	Mandatory	3	0	0	30	20	50		100		150	3
5	BCSBS0401	Operations Research	Mandatory	2	0	0	30	20	50		50		100	2
6	BCSBS0408	Design Thinking	Mandatory	2	0	0	30	20	50		50		100	2
7	BCSBS0453	Operating Systems Lab (Unix)	Mandatory	0	0	2				25		25	50	1
8	BCSBS0454	Database Management Systems Lab	Mandatory	0	0	2				25		25	50	1
9	BCSBS0452	Software Design with UML Lab	Mandatory	0	0	2				25		25	50	1
10	BCSBS0451	Operations Research Lab	Mandatory	0	0	2				25		25	50	1
11	BCSBS458	Design Thinking Lab	Mandatory	0	0	2				25		25	50	1
12	BNC0404	Essence of Indian Traditional Knowledge	Compulsory Audit	2	0	0	30	20	50		50		100	NA
		Massive Open Online Courses (For B.Tech. Hons. Degree)	MOOCs											
		TOTAL		**18**	**0**	**10**	**210**	**140**	**350**	**125**	**550**	**125**	**1050**	**21**

* List of MOOCs Based Recommended Courses for Second year (Semester-IV) B. Tech Students

S. No.	Subject Code	Course Name	University / Industry Partner Name	No of Hours	Credits
1	BMC0010	Comprehensive Training on Unix and Linux OS Fundamentals	Infosys Wingspan (Infosys Springboard)	29h 53m	2
2	BMC0058	Implementing Database Using SQL Server 2012	Infosys Wingspan (Infosys Springboard)	26h 29m	2
3	BMC0059	TechA Build and Deploy Projects Certification	Infosys Wingspan (Infosys Springboard)	14h	1

PLEASE NOTE: -

- **Compulsory Audit (CA) Courses (Non-Credit - BNC0404)**

➢ 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.

A student will be eligible to get Undergraduate degree with Honours only, if he/she completes the additional MOOCs courses such as Coursera 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 Coursera 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

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BCSBS0306Z	**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 fundamental concepts of formal languages and automata and analyze regular languages for language classification and minimization.	K2, K4
CO2	Understand context-free and context-sensitive languages and apply formal techniques to examine language ambiguity and equivalence. .	K3
CO3	Implement and analyze Turing Machine for Recursive and Recursive Enumerable Languages.	K4
CO4	Analyze decidable and undecidable problems in formal languages	K4
CO5	Perform Polynomial time reduction and proving NP-Completeness of basic NP-hard Problem.	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	3	-	2	2	-	-	-	-	-	-	3	2	-
CO2	3	3	-	2	-	-	-	-	-	-	-	3	2	2
CO3	3	2	-	-	3	-	-	-	-	-	-	3	2	2
CO4	2	3	-	3	-	-	-	-	-	-	2	2	2	2
CO5	2	2	-	-	3	-	-	-	-	2	2	3	2	2

Course Contents / Syllabus

Module 1	**Introduction of Regular Languages and Finite Automata**	**10 hours**

Alphabet, languages and grammars, productions and derivation, Chomsky hierarchy of languages. Regular expressions and languages, deterministic finite automata (DFA) and equivalence with regular expressions, nondeterministic finite automata (NFA) and equivalence with DFA, 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, minimization of finite automata.		
Module 2	**Context-free Languages and Pushdown Automata**	**10 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.		
Module 3	**Turing machines**	**10 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.		
Module 4	**Undecidability**	**10 hours**
Church-Turing thesis, Universal Turing machine, Universal and diagonalization languages, Reduction between languages and Rice’s theorem, Undecidable problems about languages		
Module 5	**Basic Introduction to Complexity**	**08 hours**
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		**48 hours**
Textbook:		

1.	Introduction to Automata Theory, Languages, and Computation	John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman.
Reference Books:		
1. 2. 3. 4. 5.	Elements of the Theory of Computation Automata and Computability Introduction to the Theory of Computation Introduction to Languages and the Theory of Computation Computers and Intractability: A Guide to the Theory of NP Completeness	Harry R. Lewis and Christos H.Papadimitriou. Dexter C. Kozen. Michael Sipser

		John Martin. M. R. Garey and D. S. Johnson.
NPTEL/ Youtube/ Faculty Video Link:		
1.	https://www.youtube.com/playlist?list=PLbRMhDVUMngcwWkzVTm_kFH6JW4JCtAUM (Lecture 1 to 32) https://www.youtube.com/results?search_query=%23AutomataTheory	
2.	https://www.youtube.com/playlist?list=PLbRMhDVUMngcwWkzVTm_kFH6JW4JCtAUM (Lecture 36 to 58) https://www.youtube.com/results?search_query=%23AutomataTheory	
3.	https://www.youtube.com/playlist?list=PLbRMhDVUMngcwWkzVTm_kFH6JW4JCtAUM(Lecture 59 & 60) https://www.youtube.com/results?search_query=%23AutomataTheory	
4.	https://www.youtube.com/watch?v=yIim-rT7jKY&list=PLyqSpQzTE6M9-v4V62bCygAVYGluaijyo (Lecture 37 to 47) https://www.youtube.com/results?search_query=%23AutomataTheory	
5.	https://www.youtube.com/watch?v=yIim-rT7jKY&list=PLyqSpQzTE6M9-v4V62bCygAVYGluaijyo (Lecture 49 to 63) https://www.youtube.com/results?search_query=%23AutomataTheory	

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BCSBS0303	**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

Module 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.

Module 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.

Module 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.

Module 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.

Module 5	**Parallel Processors, Memory organization**	**10 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	**44 hours**

Textbook:

1.	Computer System Architecture	M. Mano
2.	Computer Organization, McGraw-Hill	Carl Hamacher, Zvonko Vranesic

Reference Books:

1.	Computer Organization and Architecture- Pearson Education	William Stallings
2.	“Computer Architecture”, Oxford University Press, Eighth Impression	Behrooz Parahami

NPTEL/ YouTube/ Faculty Video Link:

1.	https://www.youtube.com/watch?v=nQnS8YAQcDY
2.	https://www.youtube.com/watch?v=DsK35f8wyUw
3.	https://onlinecourses.nptel.ac.in/noc25_cs154/preview
4.	https://www.youtube.com/watch?v=lYbGMJkqKak
5.	https://www.youtube.com/watch?v=xzNrZ_X_200

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BCSBS0301	**Course Name: Computational Statistic**	**L**	**T**	**P**	**C**
Course Offered in: B. Tech (CSBS)		**3**	**1**	**0**	**4**

Pre-requisite: Statistics & Probability, Basics of Python

Course Objectives: Course Objectives: The objective of the course is to enable the student to use modern computer intensive statistical methods as tools to investigate statistical procedures, perform inference and conduct statistical analysis using computation and other software's.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Understand the characteristics and applications of multivariate normal distribution.	K2
CO2	Understanding of discriminant analysis techniques used for classification and dimensionality reduction in supervised learning	K2
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

Module 1	**Multivariate Normal Distribution**	**10 hours**

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

Module 2	**Discriminant Analysis**	**10 hours**

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

Module 3	**Principal Component Analysis**	**10 hours**

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

Module 4	**Factor Analysis**	**10 hours**
Factor analysis model, Extracting common factors, determining number of factors, Transformation of factor analysis solutions, Factor scores.		
Module 5	**Cluster Analysis**	**08 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	**48 hours**
Textbook:		
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
NPTEL/ Youtube/ Faculty Video Link:		
1.	https://www.youtube.com/watch?v=YgExEVji7xs	
2.	https://www.youtube.com/watch?v=ImKKekAyFls	
3.	https://www.youtube.com/watch?v=hkCT-6KJAK0	
4.	https://www.youtube.com/watch?v=n3y3xLNoPk4	
5.	https://www.youtube.com/watch?v=NhimXdFenrg https://www.youtube.com/watch?v=CwjLMV52tzI https://www.youtube.com/watch?v=qg_M37WGKG8	

NIET Greater Noida	NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY GREATER NOIDA-201306 (An Autonomous Institute) School of Computer Science in Emerging Technologies

Course Code: BCSBS0302Z	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 ,C++ , and its features	K1
CO2	Apply object-oriented principles and C++ features to build modular programs using classes and objects.	K3
CO3	Apply object-oriented programs using inheritance, polymorphism, operator overloading, and generic programming with templates.	K3
CO4	Analyze and evaluate the object-oriented model by using UML diagrams.	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	1	2	1	-	1	1	1	1	3	1	1
CO2	3	3	2	1	2	1	-	2	1	1	1	3	2	1
CO3	3	3	3	2	3	1	1	2	2	2	2	3	2	2
CO4	2	2	3	2	2	1	1	3	2	2	1	3	2	1
CO5	3	2	1	1	2	1	-	1	1	1	1	3	1	1

Course Contents / Syllabus

Module 1	C Foundations & C++ Evolution	10 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		
Module 2	**The Fundamentals of OOP and OOP Facilities**	**10 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)		
Module 3	**Essentials of Object Oriented Programming and Generic Programming**	**10 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.		
Module 4	**Object Oriented Design and Modeling**	**8 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	**48 hours**
Textbook:		
1.	“Object Oriented Modeling and Design”, PHI	James Rumbaugh et. Al
2.	Object Oriented Programming with C++,	E Balagurusamy, TMH
Reference Books:		
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
NPTEL/ YouTube/ Faculty Video Link:		
1.	https://www.youtube.com/watch?v=bIzTKJzs92w	
2.	https://www.youtube.com/watch?v=pRC09Tz9iVE	
3.	https://www.youtube.com/watch?v=A38y7OO8OK4	
4.	https://www.youtube.com/watch?v=rr7HVs4d1Qo	
5.	https://www.youtube.com/watch?v=t5SuR0rbAxA&list=PLrgz73xxhUkPBK2de3CuRb7F3zKh_sqUp	

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BCSBS0304Z	**Course Name: Software Engineering**	**L**	**T**	**P**	**C**
Course Offered in: Computer Science and Business Systems		**3**	**0**	**0**	**3**

Pre-requisite: Basic knowledge about software and its types, and knowledge of any Object-Oriented programming language

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 the foundational concepts and evolution of software engineering, including life cycle models, software project management principles, and the importance of engineering approaches in large-scale software development	K2
CO2	Apply techniques for software project planning, cost estimation, risk analysis, configuration management, and software quality assurance using standard model	K3
CO3	Analyze software requirements using structured modeling methods and UML; evaluate design quality using object-oriented principles, metrics, and patterns for robust and maintainable software systems	K4
CO4	Analyze and assess test strategies using black-box and white-box testing techniques, and verify software through methods targeting functional and non-functional requirements.	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	1	1	1	-	-	-	-	1	-	3	2	2
CO2	3	2	2	1	2	1	-	-	-	2	1	3	3	3
CO3	3	3	3	2	2	-	-	-	-	2	1	3	3	3
CO4	3	2	3	2	2	1	-	-	-	2	-	3	3	3

Course Contents / Syllabus

Module 1	**Introduction**	**8 hours**

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.

Module 2	**Software Project Management & Software Quality and Reliability**	**12 hours**

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

<table>
<tr><td colspan="4">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.</td></tr>
<tr><td><b>Module 3</b></td><td colspan="2"><b>Software Requirements Analysis, Design and Construction</b></td><td><b>8 hours</b></td></tr>
<tr><td colspan="4">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.</td></tr>
<tr><td><b>Module 4</b></td><td colspan="2"><b>Object Oriented Analysis, Design and Construction & Software Testing</b></td><td><b>12 hours</b></td></tr>
<tr><td colspan="4">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.</td></tr>
<tr><td colspan="3" align="right"><b>Total Lecture Hours</b></td><td><b>40 hours</b></td></tr>
<tr><td colspan="4"><b>Textbook:</b></td></tr>
<tr><td><b>1.</b></td><td>Software Engineering, , Edition 9, Pearson</td><td colspan="2"><b>Author</b>
Ian Sommerville</td></tr>
<tr><td colspan="4"><b>Reference Books:</b></td></tr>
<tr><td><b>S.No</b></td><td>1. Fundamentals of Software Engineering,
2. Software Requirements and Specification: A Lexicon of Practice, Principles and Prejudices
3. The Unified Development Process
4. Design Patterns: Elements of Object-Oriented Reusable Software
5. Software Metrics: A Rigorous and Practical Approach
6. Software Engineering: Theory and Practice
7. Object-Oriented Software Construction, Bertrand Meyer</td><td colspan="2">Carlo Ghezzi, Jazayeri Mehdi, Mandrioli Dino
Michael Jackson
Ivar Jacobson, Grady Booch, James Rumbaugh
Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides
Norman E Fenton, Shari Lawrence Pfleeger
Shari Lawrence Pfleeger and Joanne M. Atlee
Bertrand Meyer
Ivar Jacobson</td></tr>
</table>

	8. Object Oriented Software Engineering: A Use Case Driven Approach 9. Touch of Class: Learning to Program Well with Objects and Contracts -- UML Distilled: A Brief Guide to the Standard 10. Object Modeling Language --Martin Fowler	Bertrand Meyer

NPTEL/ Youtube/ Faculty Video Link:	
Unit 1	https://youtu.be/x-jqSXYE4S
Unit 2	https://youtu.be/mGkkZoFc-4I
Unit 3	https://youtu.be/sGxgZxwuHzc
Unit 4	https://youtu.be/BNk7vni-1Bo
Unit 5	https://youtu.be/8swQr0kckZI

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

LAB Course Code: BCSBS0353	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

1. Design the control unit of a computer using hardwiring based on its RTL description.
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.
3. Implementation of Half adder and full adder
4. Implementation of Half subtractor and full subtractor
5. Implementation of array multiplier
6. Implementation of array multiplexer and demultiplexer
7. Implementation of array encoder and decoder
8. Implementation of Synchronous and Asynchronous counter
9. Implementation of Shift registers.
10. Design of an arithmetic and logic unit
11. Design of an 8-bit input/output system with four 8-bit internal register.
12. Design the data path of a computer from its registers transfer language

13. Write an assembly or C program to read data from a keyboard or another input device using program-controlled I/O.
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.
15. Write a program to communicate with a USB device, such as a flash drive or keyboard, to read or write data.
16. Write a program to simulate a basic instruction pipeline with 4 stages: Fetch, Decode, Execute, and Write Back.
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.
18. Write a program in a parallel programming language (e.g., OpenMP or MPI) to perform matrix multiplication using multiple processors.
19. Write a simulation program to model a multi-processor system where each processor has its own cache.
20. Implement a cache coherence protocol (e.g., MESI) to ensure data consistency across all caches when multiple processors access shared memory.
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.
Total Hours: 20 hrs.

NIET Greater Noida

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

LAB Course Code: BCSBS0352	**LAB Course Name: Object Oriented Programming Lab**	**L**	**T**	**P**	**Credits**
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 basic object-oriented programming concepts such as classes, objects, inheritance, and polymorphism in C++ and Java.	K3
CO2	Design and develop small-scale software projects using UML diagrams and object-oriented principles	K6
CO3	Debug and test object-oriented programs to ensure proper functionality and adherence to design specifications.	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	1	1	1	2	2	2	3	2	2
CO2	3	3	3	2	3	1	2	1	3	3	2	3	3	3
CO3	2	3	2	3	2	1	1	2	2	2	2	2	2	2

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

Sr. No.	**Program**
1.	Parameter passing: passing parameter by value vs by reference, passing array as constant pointer.
2.	Function overloading: writing string operations like strcat and strncat, strcpy and strncpy as overloaded functions.
3.	Dynamically allocating space for a pointer depending on input and doing this repeatedly, depending on different inputs and finally de-allocating the pointer
4.	Define class complex with all possible operations: constructor, destructor, copy constructor, assignment operator with the data members stored as pointer to integers.
5.	Define class vector of integers with all possible operations like constructor, destructor, copy constructor and assignment operators
6.	Define class matrix of integers with all possible operations like constructor, destructor, copy constructor and assignment operators

7.	Define class matrix of integers using vector, with all possible operations like constructor, destructor, copy constructor and assignment operators
8.	Define class stack, queue, linked-list, array, set using some data-type (int) with data members kept as private and functions kept in both protected and public sections
9.	Define class complex with all possible operators: constructor, destructor, copy constructor, assignment operator and operators >, =, <=, ==, ++ (pre and post), +, +=, ( ), with the data members stored as pointer to integers.
10.	Define class vector of integers with all possible operations like constructor, destructor, copy constructor and assignment operators>, =, <=, ==, ++ (pre and post), +, +=, ( ).
11.	Define class matrix of integers with all possible operations like constructor, destructor, copy constructor and assignment operators>, =, <=, ==, ++ (pre and post), +, +=, ( ).
12	Define class matrix of integers using vector, with all possible operations like constructor, destructor, copy constructor and assignment operators>, =, <=, ==, ++ (pre and post), +, +=, ( ).
13	Define stack and queue inherited from array class, with standard functions and operators.
14	Define a class called ‘array’ with data type passed as template type with constructor, destructor, copy constructor and assignment operators and index operator
15	Define template functions for compare and use it in the algorithms like bubble sort, insertion sort, merge sort
16	Write a C++ program that demonstrates how to format output using manipulators like setw, setprecision, and fixed. Display a table of numbers with different formats (e.g., right-aligned, fixed-point notation)
17	Implement a program that reads different data types (e.g., integer, float, string) from the user. Use input manipulators like to handle whitespaces and getline for reading entire lines. Demonstrate how these manipulators affect the input operation
18	Create a class Complex to represent complex numbers. Overload the << and >> operators to enable formatted input and output of complex numbers. Write a program to read and display a complex number using these overloaded operator
19	Define class model for complex number, student class, book class and show it using UML diagram as well as concrete class
20	Show behavioral modeling through sequence diagram and activity diagram for workflow in a typical log-in, log-out situation.
21	Design a program that combines the use of manipulators and overloaded operators to format and display a list of student records (name, roll number, and grade) in a tabular format
22	Write a C++ program that defines a custom manipulator to format dates in a specific style (e.g., DD-MMYYYY). Demonstrate the use of this custom manipulator with input and output streams.

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

LAB Course Code: BCSBS0351	**LAB Course Name: Computational Statistic Lab**	**L**	**T**	**P**	**C**
Course Offered in B.Tech. CSBS		**0**	**0**	**4**	**2**

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

Course Objectives: To develop and understand the modern computational statistical approaches and their applications to different datasets.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	To understand the Python programs using normal Functions, Numeric Types, Sequences and Class Definition, Constructors, Text & Binary Files.	K2
CO2	Understanding the concepts of Data Wrangling includes Reshaping and Pivoting, Data Transformation, String Manipulation, Regular Expressions.	K2
CO3	Understand the visualization of Data Aggregation, Group Operations and Time series.	K3
CO4	Understand the implementation of Data Visualization in Python includes Plotting Graphs, Controlling Graph, Adding Text, More Graph Types	K3

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

CO-PO Mapping	**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO 1**	**PSO 2**	**PSO 3**
CO1	2	2	3	1	3	1	-	-	-	-	-	1	2	2
CO2	2	3	3	3	3	1	-	-	-	-	-	1	2	2
CO3	2	2	1	2	1	1	-	-	-	-	-	2	3	3
CO4	3	3	3	3	3	1	-	-	-	-	-	2	3	3

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

1. Creation and manipulation of Vectors, Matrices, Arrays, Lists, Factors and Data Frames
2. Create a program to adding element of the matrix in a python
3. Compute Estimators of the main statistical measures like Mean, Variance, Standard Deviation, Covariance, Correlation and Standard error with respect to any example. Display graphically the distribution of samples.
4. Install of Packages and scripts for Importing and Exporting Data
5. Plot the Normal Distribution for class test result of a particular subject. Identify the Skewness and Kurtosis
6. Visualize Statistical Graphs using Scatter Plots, Box Plots, Histograms, Pie diagram
7. Write a program to compute summary statistics such as standard
8. Write a program to demonstrate Regression analysis

9. Load the dataset:

birthwt Risk Factors Associated with Low Infant Birth Weight at https://raw.github.com/neurospin/pystatsml/master/datasets/birthwt.csv

a. Test the association of mother’s (bwt) age and birth weight using the correlation test and linear regression.

b. Test the association of mother’s weight (lwt) and birth weight using the correlation test and linear regression.

c. Produce two scatter plots of: (i) age by birth weight; (ii) mother’s weight by birth weight.

10. Perform clustering of the iris dataset based on all variables using Gaussian mixture models. Use PCA to visualize clusters.

Total Hours: 48 hrs.

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY
GREATER NOIDA-201306
(An Autonomous Institute)
School of Computer Science in Emerging Technologies

LAB Course Code: BCSBS0354Z	**LAB Course Name: Software Engineering Lab**	**L**	**T**	**P**	**C**
Course Offered in: Computer Science and Business Systems		**0**	**0**	**4**	**2**

Pre-requisite:

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**	**1**	**1**	**1**	**2**	**2**	**2**	**1**	3	2	2
CO2	**3**	**3**	**3**	**2**	**2**	**1**	**1**	**2**	**2**	**2**	**1**	3	3	3
CO3	**3**	**2**	**2**	**2**	**1**	**2**	**2**	**3**	**2**	**3**	**2**	3	2	3

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

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

2. Develop function-oriented design using SA/SD methodology.

3. Develop object-oriented design using UML.

4. Designing and implementing test cases manually.

5. Designing and implementing test cases automatically using a tool.

6. Use of appropriate CASE tools and other tools (any one) such as configuration management tools, program analysis tools in the software life cycle.

7. Create a Software Design Document (SDD): Object and Class diagram.

8. Create Interaction diagram: sequence diagram, collaboration diagram for SDD.

9. Design test suite for equivalence class partitioning

10.Mini Project with CASE tools

Total Hours: 20 hrs.

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BNC0303	Course Name: Indian Constitution	L	T	P	C
Course Offered in: CSBS		2	0	0	0

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

Course Objectives: To understand the modalities about Indian Constitution & apply different laws and regulations to engineering practices.

Course Outcome: After completion of the course, the student will be able to:		Bloom's Knowledge Level (KL)
CO1	Identify and explore the basic features and modalities about Indian constitution.	K1
CO2	Understand and relate the functioning of Indian parliamentary system at the center and state level.	K2
CO3	Apply different aspects of Indian Legal System and its related bodies	K3
CO4	Discover and apply different laws and regulations related to engineering practices.	K4
CO5	Correlate role of engineers with different organizations and 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	2	2	-	2	2	3	-	2	-	1	-	2	2	2
CO2	2	3	1	1	2	3	-	2	1	1	1	2	2	1
CO3	3	2	3	2	2	2	1	2	1	2	2	3	2	2
CO4	2	3	-	3	3	1	1	2	1	1	1	2	2	3
CO5	2	2	2	2	3	1	-	2	2	2	1	3	2	2

Course Contents / Syllabus

Module 1	**Introduction and Basic Information about Indian Constitution**	**6 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.

<table>
<tr><td><b>Module 2</b></td><td><b>Union Executive and State Executive</b></td><td><b>6 hours</b></td></tr>
<tr><td colspan="3">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.</td></tr>
<tr><td><b>Module 3</b></td><td><b>Introduction and Basic Information about Legal System</b></td><td><b>6 hours</b></td></tr>
<tr><td colspan="3">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.</td></tr>
<tr><td><b>Module 4</b></td><td><b>Intellectual Property Laws and Regulation to Information & Business Organizations and E Governance</b></td><td><b>12 hours</b></td></tr>
<tr><td colspan="3">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.</td></tr>
<tr><td colspan="2"><b>Total Lecture Hours</b></td><td><b>30 hours</b></td></tr>
</table>

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.	Intellectual Property Law, Eastern Law House, New Delhi	P. Narayan

NPTEL/ Youtube/ Faculty Video Link:

NPTEL/ Youtube/ Faculty Video Link:	
1.	https://www.youtube.com/watch?v=z-UzdY-pXTc
2.	https://www.youtube.com/watch?v=m6E83v_wzGg
3.	https://www.youtube.com/watch?v=dBo9SzzbugA
4.	https://www.youtube.com/watch?v=zkWJAvg6_ME

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BCSBS0403	**Course Name: Operating Systems**	**L**	**T**	**P**	**C**
Course Offered in: B. Tech (CSBS)		**3**	**0**	**0**	**3**

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

Module 1	**Introduction**	**10 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.

Module 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		
Module 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.		
Module 4	**Memory Management**	**10 hours**
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		
Module 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.		
	Total Lecture Hours	**48 hours**
Textbook:		
1.	Operating System Concepts Essentials.	Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, Edition 8
Reference Books:		

1.	Operating Systems: Internals and Design Principles	William Stallings, Edition 9
2.	Operating System: A Design-oriented Approach	Charles Patrick
3.	Operating Systems: A Modern Perspective	Gary J. Nutt.
4.	Design of the Unix Operating Systems	Maurice J. Bach.
5.	Understanding the Linux Kernel	Daniel Pierre Bovet, Marco Cesa

NPTEL/ Youtube/ Faculty Video Link:

1.	https://www.youtube.com/watch?v=jciGIvn7UfM&list=PLyqSpQzTE6M9SYI5RqwFYtFYab94gJpWk&index=1 https://www.youtube.com/watch?v=2i2N_Qo_FyM&t=134s https://www.youtube.com/watch?v=ucVm_arB-fw https://www.youtube.com/watch?v=lmlLdGMrdGU
2.	https://www.youtube.com/watch?v=4hCih9eLc7M https://www.youtube.com/watch?v=TGpSBceX36E https://www.youtube.com/watch?v=Y1PF0fE-v9M
3.	https://www.youtube.com/watch?v=fYv_sLu-1uE&list=PLyqSpQzTE6M9SYI5RqwFYtFYab94gJpWk&index=25 (Lecture 24-33) https://www.youtube.com/watch?v=UczJ7misUEk https://www.youtube.com/watch?v=_IxqinTs2Yo
4.	https://www.youtube.com/watch?v=_pKbqFyG03s https://www.youtube.com/watch?v=kQKpJ4bD8TA https://www.youtube.com/watch?v=dReNOOVZAkk https://www.youtube.com/watch?v=Ev4BET3i5R0
5.	https://www.youtube.com/watch?v=U1Jpvni0Aak https://www.youtube.com/watch?v=4svECqoC6Do https://www.youtube.com/watch?v=S4XYOjSoAQE

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BCSBS0404	**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	Analyze database used to solve real world and complex problem and design the ER, EER Model.	K4
CO2	Analyze and apply Structured Query Language (SQL) or Procedural Query Language (PL/SQL) to solve the complex queries. Implement relational model, integrity constraints.	K4
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

Module 1	**Introduction**	**08 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.

Module 2	**Relational Query Languages**	**08 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.		
Module 3	**Query Processing and Optimization**	**08 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.		
Module 4	**Transaction Processing**	**08 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.		
Module 5	**Database Security**	**08 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	**40 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
NPTEL/ Youtube/ Faculty Video Link:		
Unit 1	https://www.youtube.com/watch?v=hDcJnsYQiEc	
Unit 2	https://www.youtube.com/watch?v=MYXIw5aqYNQ	

Unit 3	https://www.youtube.com/watch?v=4YAsAdCa9sU
Unit 4	https://www.youtube.com/watch?v=mN7tHGBbg_Y
Unit 5	https://www.youtube.com/watch?v=B7tTQ272OHE

NIET Greater Noida

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY
GREATER NOIDA-201306
(An Autonomous Institute)
School of Computer Science in Emerging Technologies

Course Code: BCSBS0402Z	**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 and Evaluate the conceptual model into various scenarios and applications.	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	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

Module 1	**Software development process**	**10 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.

Module 2	**Introduction to the UML Language**	**8 hours**

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

Module 3	**Requirement s Analysis Using Case Modeling**	**8 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.		
Module 4	**The Logical View Design Stage**	**10 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.		
Module 5	**Models**	**10 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	**46 hours**

Textbook:

1. **2.**	The Unified Modelling Language User Guide, Pearson Education Object-Oriented Software Engineering: using UML, Patterns, and Java.	Grady Booch, James Rumbaugh Erich Gamma, Richard Helm

Reference Books:

1.	Design Patterns: Elements of Reusable Object-Oriented Software.	Erich Gamma, Richard Helm, Ralph Johnson, and John M.

NPTEL/ Youtube/ Faculty Video Link:

1.	https://www.youtube.com/watch?v=kSU2MPeptpM
2.	https://www.youtube.com/watch?v=WnMQ8HlmeXc
3.	https://www.youtube.com/watch?v=azTLDkiqGVk&list=PLbRMhDVUMngf8oZR3Dp K MvYhZKga90JVt&index=37 https://www.youtube.com/watch?v=l9XFipXoJb0&list=PLbRMhDVUMngf8oZR3DpK Mv YhZKga90JVt&index=15
4.	https://www.youtube.com/watch?v=9KokDbcr6cM&list=PLbRMhDVUMngf8oZR3Dp KMvYh ZKga90JVt&index=36 https://www.youtube.com/watch?v=7Pc5-birfmk&list=PLbRMhDVUMngf8oZR3DpKMvYhZKga90JVt&index=35
5.	https://www.youtube.com/watch?v=sPORiupW4mw

NIET Greater Noida	**NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY** **GREATER NOIDA-201306** **(An Autonomous Institute)** **School of Computer Science in Emerging Technologies**

Course Code: BCSBS0405Z	**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 Management

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		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	3	3	2	2	2	-	-	-	-	-	-	2	2	2
CO2	3	3	3	3	2	-	-	2	2	1	-	2	2	2
CO3	3	3	3	3	2	-	1	2	2	1	2	2	2	2
CO4	3	3	3	3	2	2	1	2	2	1	2	2	2	3
CO5	3	3	3	3	2	3	2	-	3	3	2	2	2	3

Course Contents / Syllabus

Module 1	**Innovation**	8 Hours
Innovation: What and why? Innovation as a core business process, Sources of innovation, Knowledge push vs. need pull innovations.		
Module 2	Building an Innovative Organization	8 Hours
Creating new products and services, Go- it- alone approach Exploiting open innovation and collaboration, Use of innovation for starting a new venture		
Module 3	Entrepreneurship	8 Hours
Opportunity recognition and entry strategies, Entrepreneurship as a Style of Management, Maintaining Competitive Advantage- Use of IPR to protect Innovation.		
Module 4	Financial Planning	8 Hours

<table>
<tr><td colspan="6">Financial Projections and Valuation, Stages of financing, Debt, Venture Capital and other forms of Financing.</td></tr>
<tr><td colspan="3"><b>Module 5</b></td><td colspan="2">Intellectual Property Rights (IPR)</td><td>8 Hours</td></tr>
<tr><td colspan="6">Introduction and the economics behind the development of IPR: Business Perspective, IPR in India – 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?</td></tr>
<tr><td colspan="5" align="right"><b>Total Lecture Hours</b></td><td>48 Hours</td></tr>
<tr><td colspan="6"><b>Textbook:</b></td></tr>
<tr><td colspan="2">1.</td><td colspan="2">Joe Tidd, John Bessant. Managing Innovation: Integrating Technological, Market and Organizational Change</td><td colspan="2">Joe Tidd, John Bessant.</td></tr>
<tr><td colspan="6"><b>Reference Books:</b></td></tr>
<tr><td colspan="2">1.</td><td colspan="2">Case Study Materials: To be distributed for class discussion</td><td colspan="2"></td></tr>
<tr><td colspan="6"><b>NPTEL/ YouTube/ Faculty Video Link:</b></td></tr>
<tr><td>1.</td><td colspan="5">https://youtu.be/YtDh0J3m8JY, https://www.youtube.com/watch?v=-PjqIO45cJY&t=292s</td></tr>
<tr><td>2.</td><td colspan="5">https://www.youtube.com/watch?v=SXoRGYz8wl0&t=1s</td></tr>
<tr><td>3.</td><td colspan="5">https://www.youtube.com/watch?v=BATuWajjgLE&t=600s
https://www.youtube.com/watch?v=ntXxBHWMKfo&t=2s</td></tr>
<tr><td>4.</td><td colspan="5">https://www.youtube.com/watch?v=1qy1GX6gugw&t=9s</td></tr>
<tr><td>5.</td><td colspan="5">https://www.youtube.com/watch?v=pqRqJ3e4PUE&t=1s
https://www.youtube.com/watch?v=CElNe46m2iU</td></tr>
</table>

NIET Greater Noida

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BCSBS0401	**Course Name: Operations Research**	**L**	**T**	**P**	
Course Offered in: B. Tech. (CSBS)		**3**	**0**	**0**	

Pre-requisite: This course aims to provide basic knowledge of mathematics about Industry Requirement of Subject (mention the companies where this subject can play a role or this subject is required for the mentioned job role)

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

Course Outcome:		Bloom's Knowledge Level (KL)
CO1	Understand the characteristics of different types of decision-making environments and the appropriate decision-making approaches in real world problems.	K2
CO2	Apply linear programming problem to business and industry.	K3
CO3	Apply the Transportation problem and Assignment problem to real life problems.	K3
CO4	Apply project management concepts like CPM, PERT and inventory Control to reduce cost and time in manufacturing and production.	K3
CO5	Apply the concept of Simulation Methodology to real life applications.	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	2
CO2	3	3	1	3	-	-	-	-	-	-	1	3	2	3
CO3	3	2	1	3	-	-	-	-	-	-	2	3	2	3
CO4	3	3	1	3	-	-	-	-	-	-	1	3	2	3
CO5	3	2	1	2	-	-	-	-	-	-	1	3	2	3

Course Contents / Syllabus

Module 1	**Origin of OR**	**10 hours**

Origin of OR and its definition. Concept of optimizing performance measure, Types of OR problems, Deterministic vs. Stochastic optimization, Phases of OR problem approach – problem formulation, building mathematical model, deriving solutions, validating model, controlling and implementing solution.

Module 2	**Linear programming**	**10 hours**

Linear programming – Examples from industrial cases, formulation & definitions, Matrix form. Implicit assumptions of LPP.

Some basic concepts and results of linear algebra – Vectors, Matrices, Linear Independence/Dependence of vectors, Rank, Basis, System of linear eqns., Hyperplane, Convex set, Convex polyhedron, Extreme points, Basic feasible solutions.

Module 3	**Transportation Problem & Assignment Problem:**	**10 hours**

Transportation Problem: - Examples, Definitions – decision variables, supply & demand constraints, formulation, Balanced & unbalanced situations, Solution methods – NWCR, minimum cost and VAM, test for optimality (MODI method), degeneracy and its resolution.

Assignment Problem: - Examples, Definitions – decision variables, constraints, formulation, Balanced & unbalanced situations, Solution method – Hungarian Method.

Module 4	**PERT – CPM & Inventory Control**	**10 hours**

PERT – CPM: Project definition, Project scheduling techniques – Gantt chart, PERT & CPM, Determination of critical paths, Estimation of Project time and its variance in PERT using statistical principles, Concept of project crashing/time-cost trade-off.

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)

Module 5	**Simulation Methodology**	**08 hours**

Simulation Methodology:

Definition and steps of simulation, random number, random number generator, Discrete Event System Simulation – clock, event list, Applications of simulation

Total Lecture Hours	**48 hours**

Textbook:

1.	Operations Research: An Introduction.	H.A. TAHA
2.	Operations Research	S. KALAYATHY

Reference Books:

1.	K.G. Murthy, Linear Programming.	K..G. Murthy
2.	G. Hadley, Linear Programming.	G. Hadley
3.	Introduction to Operations Research.	F.S. Hiller and G.J. Lieberman
4.	Operations Research and Management Science	A. Ravi Ravindran

NPTEL/ Youtube/ Faculty Video Link:

1.	https://www.youtube.com/watch?v=WrAf6zdteXI
2.	https://www.youtube.com/watch?v=JxnPBrNccqY
3.	https://www.youtube.com/watch?v=J1WwNKDdDC0
4.	https://www.youtube.com/watch?v=v2FT9PoFJ9Y
5.	https://www.youtube.com/watch?v=9qnLpjpnsuQ

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

Course Code: BCSBS0408	**Course Name: Design Thinking**	**L**	**T**	**P**	**C**
Course Offered in: B. Tech. CSBS		**2**	**0**	**0**	**2**

Pre-requisite: None

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	Identify 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	K3
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	2	1	1	2	1	2	2	2	2	2	2	**3**
CO2	1	2	1	1	1	1	2	2	2	1	2	2	2	2
CO3	1	2	1	2	2	2	1	2	2	1	2	2	3	3
CO4	1	2	1	1	1	1	1	1	1	1	2	2	2	2
CO5	1	2	1	1	1	1	1	1	1	1	2	2	3	2

Course Contents / Syllabus

Module 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

Module 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

Module 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

Module 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

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

NPTEL/ Youtube/ Faculty Video Link:

1	https://youtu.be/rUUuhnLkJ2s?si=_XCHnDbt_U1z0Frx https://www.youtube.com/watch?v=ldYzbV0NDp8 https://www.youtube.com/watch?v=0Fi83BHQsMA
2	https://www.youtube.com/watch?v=q654-kmF3Pc https://swayam.gov.in/nd1_noc19_mg60/preview
3	https://www.udemy.com/course/design-thinking-for-beginners/ https://www.interaction-design.org/literature/article/personas-why-and-how-you-should-use-them
4	https://www.d-thinking.com/2021/07/01/how-to-use-storytelling-in-design-thinking/ https://www.youtube.com/watch?v=GsF4reOqsGE

NIET Greater Noida

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

LAB Course Code: BCSBS0453	LAB Course Name: Operating Systems Lab (Unix)	L	T	P	C
Course Offered in: B.Tech.		**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 basic CPU scheduling and memory management algorithms.	K3
CO2	Develop and apply algorithms for process synchronization, deadlock handling, and memory/page management.	K3,K6
CO3	Demonstrate understanding and application of file systems, Unix/Linux shell programming, and system commands.	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	2	2	2	3	3	2
CO2	3	3	3	2	3	1	1	2	2	2	2	3	3	3
CO3	3	3	2	2	3	1	1	2	2	2	3	2	3	3

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

Sr. No.	Program
1.	Implement FCFS CPU Scheduling algorithm.
2.	Implement the given CPU Scheduling algorithm a) SJF b) Priority Based
3.	Implement PRIORITY CPU Scheduling Algorithm (For both Pre-emptive and nonpre-emptive).
4.	Implement Round-Robin CPU Scheduling Algorithm
5.	Implement the Producer–consumer problem using semaphores.
6.	Execute an algorithm for deadlock detection.
7.	Implement Banker's algorithm of Deadlock Avoidance
8.	Implement Contiguous memory fixed size partition scheme
9.	Implement Contiguous memory variable size partition scheme
10.	Simulate the First-Fit contiguous memory allocation technique.
11.	Simulate the Best-Fit contiguous memory allocation technique.
12.	Simulate the Worst-Fit contiguous memory allocation technique

13.	Implement the Non Continuous Memory Allocation by using Paging.
14.	Write a Program to simulate the FIFO page replacement algorithm
15	Write a Program to simulate the LRU page replacement Algorithm
16	Write a Program to simulate the Optimal page replacement Algorithm.
17	Write a Program to simulate the FCFS Disk Scheduling Algorithm.
18	Write a Program to simulate the SSTF Disk Scheduling Algorithm.
19	Implement SCAN and C-SCAN Disk Scheduling Algorithms.
20	Implement LOOK and C-LOOK Disk Scheduling Algorithms.
21	Design an algorithm and implement to organize the file using the single-level directory
22	Write a program to organize the file using two-level directories.
23	Write a C program to Sequential files for processing the student information
24	Write a C program for random access files for processing the employee details
25	Execute Various types of Linux Commands (Miscellaneous, File oriented, Directory oriented)
26	Execute a shell program, which accepts the name of a file from standard input and performs the File Readable test on it.
27	Design and execute the code to accept the name of a file from standard input and performs the File Writable test on it
28	Implement a shell program, which accepts the name of a file from standard input and performs the File executable test on it.
29	Implement Linux Networking Commands: ipconfig, traceroute, tracepath, ping, host, hostname, iwconfig.
30	Implement the following system admin commands in Linux: man, uptime, users, service, pkill, ps
31	Implement the following in Unix: a) Process creation, b) Sleep Command c) Sleep command using getpid.
32	Analyse system calls of unix operating system (fork and exit)
33	Implement Unix commands for a) Signal handling using kil, b) Wait command, c)top
34	Write a program to simulate UNIX commands like cp, ls, and grep
35	Implement Unix Shell programming for concatenation of two strings
36	Implement Unix Shell programming for a) Comparison of two strings b) Maximum of three numbers.
37	Implement Unix Shell programming for Fibonacci series
38	Write a program in Unix to whether the given year is a) a leap year or not b) Arithmetic operation using cases.
39	Write a program in Unix for factorial of a number.
40	Write a program in Unix to swap the two integers
41 1.	Write a program in Unix to whether the given number is prime or not.
	Total Hours: 48 hrs.

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

LAB Course Code: BCSBS0454	**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**
1.	Creating Entity-Relationship Diagram using case tools with Identifying (entities, attributes, keys and relationships between entities, cardinalities, generalization, specialization etc.)
2.	I. Implement DDL commands –Create, Alter, Drop etc. II. Implement DML commands- Insert, Select, Update, Delete
3.	I. Implement DCL Commands-Grant and Revoke
4.	II. Implement TCL commands- Rollback, Commit, Save point
5.	Practice Queries using COUNT, SUM, AVG, MAX, MIN, GROUP BY, HAVING, VIEWS Creation and Dropping.

	6.	Practicing Queries using ANY, ALL, IN, EXISTS, NOT EXISTS, UNION, INTERSECT, CONSTRAINTS etc.		
	7.	Practicing Sub queries (Nested, Correlated) and Joins (Inner, Outer and Equi).		
	8.	Practicing on Triggers - creation of trigger, Insertion using trigger, Deletion using trigger, Updating using trigger		
	9.	Procedures- Creation of Stored Procedures, Execution of Procedure, and Modification of Procedure.		
	10.	Cursors- Declaring Cursor, Opening Cursor, Fetching the data, closing the cursor.		
	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)		
		Total Hours: 20 hrs.		

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

LAB Course Code: BCSBS0452	**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 modeling, structural and behavioral diagrams, and system documentation, in the context of software development projects.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
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.	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	2	3	2	2	-	-	-	-	2	-	2	3	2	2
CO2	2	2	3	-	3	-	-	-	2	-	2	2	3	2
CO3	3	2	3	2	3	-	-	-	3	1	3	3	3	3
CO4	-	-	-	-	-	-	-	-	-	-	-	-	-	-
CO5	-	-	-	-	-	-	-	-	-	-	-	-	-	-

List Of Practical's

1. Use Case Diagram.
2. Use Case Diagram for ATM.
3. Use Case Diagram for Stock Maintenance System
4. Use Case Diagram for Remote Procedure Call
5. Class Diagram
6. Class Diagram for ATM
7. Class Diagram for Stock Maintenance System
8. Object Diagram
9. Object Diagram for ATM
10. Object Diagram for Stock Maintenance System
11. Sequence Diagram

12. Sequence Diagram for ATM
13. Sequence Diagram for Stock Maintenance System
14. Collaboration Diagram
15. Collaboration Diagram for ATM
16. Collaboration Diagram for Stock Maintenance System
17. Collaboration Diagram for Remote Procedure Call
18. State Chart Diagram
19. State Chart Diagram for ATM
20. State Chart Diagram for Stock Maintenance System
21. Activity Diagram
22. Activity Diagram for ATM
23. Activity Diagram for Stock Maintenance System
24. Component Diagram
25. Component Diagram for ATM
26. Component Diagram for Stock Maintenance System
27. Deployment Diagram
28. Deployment Diagram for ATM
29. Deployment Diagram for Stock Maintenance System
Total Hours: 20 hrs

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY
GREATER NOIDA-201306
(An Autonomous Institute)
School of Computer Science in Emerging Technologies

LAB Course Code: BCSBS0451	**LAB Course Name: Operations Research Lab**	**L**	**T**	**P**	**Credit**
Course Offered in: B. Tech (CSBS)		**0**	**0**	**2**	**1**

Pre-requisite: Knowledge of Mathematics Fundamentals

Course Objectives: To develop and understand the modern computational statistical approaches and their applications to different datasets.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Apply the characteristics of different types of decision-making environments and the appropriate decision-making approaches and tools to be used in each type.	K3
CO2	To find optimal solution of linear programming problem by applying graphical method and simplex method.	K3
CO3	To Solve Transportation problems and Assignment problems.	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	2
CO2	**3**	**3**	**1**	**3**	-	-	-	-	-	-	-	**2**	2	3
CO3	**3**	**2**	**1**	**3**	-	-	-	-	-	-	-	**3**	1	1

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

1. Solution of linear programming problem using graphical method with:
2. Enumeration of all basic solutions for linear programming problem.
3. Solution of linear programming problem with simplex method.
4. To solve LPP by using two phase method.
5. To solve LPP by using Big M method.
6. Solution on primal problem as well as dual problem
7. Solution based on dual simplex method.
8. Solution of transportation problem by using NWCR.
9. Solution of transportation problem by using VAM.
10. Solution of assignment problem.

Total Hours: 48 hrs.

NIET Greater Noida

Autonomous Institute

NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY

GREATER NOIDA-201306

(An Autonomous Institute)

School of Computer Science in Emerging Technologies

LAB Course Code: BCSBS458	**LAB Course Name: Design Thinking Lab**	**L**	**T**	**P**	**C**
Course Offered in: B. Tech CSBS		**0**	**0**	**2**	**1**

Pre-requisite: none

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	Identyfy 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

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	2	1	1	2	1	2	2	2	2	2	2	3
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	2	2	3

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

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.)

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

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)
4. **Ideation games Game** 1: Six Thinking Hats, Game 2: Million-dollar idea
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.
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.
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
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.
Total Hours: 20 hrs.

NIET Greater Noida	**NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY** **GREATER NOIDA-201306** **(An Autonomous Institute)** **School of Computer Science in Emerging Technologies**

Course Code: BNC0404	**Course Name: Essence of Indian Traditional Knowledge**	**L**	**T**	**P**	**Credits**
Course Offered in: B. Tech. (CSBS)		**2**	**0**	**0**	**2**

Pre-requisite: Students must have a basic understanding of human resource management.

Course Objectives: This introductory course on Human Resource Management will familiarize the students with the basic concepts, roles, functional areas, and activities of HR and help students understand the organization's employees, their interest, motivation, and satisfaction, and their belief in fair treatment- all of which impact the firm's current performance and sustainability in the long run.

	Course Outcome: After completion of the course, the student will be able to	Bloom's Knowledge Level (KL)
CO1	Understanding in-depth knowledge about human resource management.	K2
CO2	Apply the strategies on HR to gain a competitive advantage over its competitors.	K3
CO3	Understand the various effective sources and techniques for recruitment and selection of employees.	K2
CO4	Analyze and forecast the need for Human Resource Planning	K4
CO5	Understand the dimensions of Strategic HRM.	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	**3**	**2**	**-**	**2**	**2**	**3**	**-**	**2**	**-**	**1**	**-**	**2**	**2**	**-**
CO2	**2**	**3**	**1**	**1**	**2**	**3**	**-**	**2**	**1**	**1**	**1**	**2**	**2**	**2**
CO3	**3**	**2**	**3**	**2**	**3**	**2**	**1**	**2**	**1**	**2**	**2**	**3**	**2**	**2**
CO4	**2**	**3**	**-**	**3**	**3**	**1**	**-**	**2**	**1**	**1**	**1**	**2**	**2**	**3**
CO5	**2**	**2**	**2**	**2**	**3**	**1**	**-**	**2**	**2**	**2**	**1**	**3**	**2**	**3**

Course Contents / Syllabus

Module 1	**Society State and Polity in India**	**7 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.

Module 2	**Indian Literature, Culture, Tradition, and Practices**	**7 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.

Module 3	**Indian Religion, Philosophy, and Practices**	**8 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.		
Module 4	**Science, Management, Indian Knowledge System, Cultural Heritage and Performing Arts**	**8 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 Pre-colonial 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**
Textbook:		
1. 2.	Gary Dessler& Biju Varkkey, Human Resource Management, Pearson Edwin B. Flippo, Personnel Management, Tata McGraw HillSharma, R.	
Reference Books:		
1. 2.	V.S.P. Rao, Human Resource Management, Excel, India RS Dwivedi, HRD in Indian Companies, Mc Millan C.B. Memoria, Personnel Management, Himalaya	

NPTEL/ Youtube/ Faculty Video Link:	
1.	https://youtu.be/yXiSNNfyBeE?si=tmTwT3Dwvz5vfG_V
2.	https://youtu.be/VVpq6mO_WGk?si=VIx6AC0uwnASQKb6
3.	https://youtu.be/_YzG4FRn7GA?si=i2Z055SXQaa_CRex
4.	https://www.youtube.com/live/gRHgFC5iwWA?si=InVg2T8f1ObuO4Un