University of Engineering & Management, Kolkata
University of Engineering & Management, Jaipur
Institute of Engineering & Management, Kolkata

B.Tech in CSE, IT, and Allied Branches

COURSE STRUCTURE

Fifth Semester

Batch: 2024-2028

Semester V (Third Year) Curriculum								
Sl. No	Type of course	Course Code	Course Name	Hours per week				Credit Points
				Lecture	Tutorial	Practical	Sessional	
Theory Papers								
1	Engineering Science Course	ESC501	Signals & Systems	3	0	0	0	3
2	Professional Core Course	PCCCS501	Database Management Systems	3	0	0	0	3
3	Professional Core Course	PCCCS502	Theory of Computations	3	0	0	0	3
4	Professional Core Course	PCCCS503	Operating Systems	3	0	0	0	3
5	Professional Core Course	PCCCS504	Software Engineering and Project Management	3	0	0	0	3

6	Humanities &Social Sciences including Management course	ESPCS501	Essential Studies for Professionals (CS) – V	2	0	0	0	0.5
7	Mandatory Course	MCC571	Constitution of India	1	0	0	0	1
	Total			**18**	**0**	**0**	**0**	**16.5**
Practical Papers								
1	Professional Core Course	PCCCS591	Database Management Systems Laboratory	0	0	4	0	2
2	Professional Core Course	PCCCS593	Operating Systems Laboratory	0	0	4	0	2
3	Professional Core Course	PCCCS594	Software Engineering and Project Management Laboratory	0	0	4	0	2
	Total			**0**	**0**	**12**	**0**	**6**
Sessional Papers								
1	Humanities & Social Sciences including Management course	SDP581	Skill Development for Professionals - V	0	0	0	2	0.5
2	Innovative Project	PRJCS581	Innovative Project – III	0	0	0	0	1
3	Professional Core Course	PCCCS581	Quantum Computing	0	0	0	4	2
4	Professional Core Course	PCCCS582	Neural Network & Deep Learning	0	0	0	4	2
	Total			**0**	**0**	**0**	**10**	**5.5**

Mandatory Requirements								
Sl. No	**Type of course**	**Course Code**	**Course Name**	**Hours per week**				**Score/Credit /Count**
1	Co-curricular & Extra Curricular Activities	MAR	Mandatory Additional Requirements (Score)	-	-	-	-	-
2	Honours	MOOCs	Massive Open Online Course (Credit)	-	-	-	-	-
3	Certification	IFC	Industry and Foreign Certification (Count)	-	-	-	-	-
Total				**18**	**0**	**12**	**10**	**28**

University of Engineering & Management, Kolkata
University of Engineering & Management, Jaipur
Institute of Engineering & Management, Kolkata

B.Tech in Computer Science and Engineering

DETAILED SYLLABUS

Course Code- ESC501
Course Title – Signals & Systems
Credit – 3
Category – Engineering Science Course
Semester – V
L:T:P:S – 3:0:0:0

Pre-requisite –
(1) Concepts of basic electrical and electronics circuits.
(2) Knowledge in algebra and calculus with problem-solving capability (studied in Mathematics).
(3) Fundamental concepts on various transformations (studied in Mathematics)

Course Outcomes:

CO1	**Understand the concepts of continuous time and discrete time signals.**
CO2	**Evaluate the frequency spectra for different kinds of signals.**
CO3	**Analyze different transformations for continuous and discrete signals.**
CO4	**Design sampling frequency and filters to recover the original signal**

Study Material	Coursera	NPTEL	Linkedin Learning	Infosys Springboard

Module No.	Topic	Sub-topics	Mapping with Industry and International Academia	Lecture Hours	Corresponding Lab Assignments	Textbook Mapping
1	**Introduction to Signals and Systems**	Signals and systems are seen in everyday life and various branches of engineering and science. Energy and power signals, continuous and discrete time signals, continuous and discrete amplitude signals. System properties: linearity: additivity and homogeneity, shift- invariance, causality, stability, realizability. Linear shift-invariant (LSI) systems, impulse response and step response, convolution, input- output behavior with aperiodic convergent inputs. Characterization of causality and stability of linear shift invariant	***International Academia:*** (MIT Open Courseware): ***https://ocw.mit.edu/courses/6-003-signals-and-systems-fall-2011/pages/lecture-notes/*** ***AICTE-prescribed syllabus:*** https://www.aicte-india.org/sites/defa	9	1. Simulation of different signals using MATLAB 2. Different operations on signals. 3. Introduction to programming using MATLAB	**Signals and Systems,P. Ramesh Babu, R. Anandanatarajan** Chapter 1 (Pages 1.1 - 1.85) Chapter 2 (Pages 2.1 - 2.30 & 2.40 - 2.41)

		systems. System representation through differential equations.	ult/files/Model_Curriculum/AICTE%20-%20UG%20CSE.pdf ***Industry Mapping:*** *MATLAB, SCILAB, OCTAVE*			
2	**Signal operation of LTI systems**	Periodic and Aperiodic inputs to an LSI system, the notion of a frequency response and its relation to the impulse response. Convolution. Correlation. Fourier series representation with evaluation of the coefficient. Relation between trigonometric & Exponential Fourier Series. Gibb's Phenomenon.	***International Academia:*** (MIT Open Courseware): ***https://ocw.mit.edu/courses/6-003-signals-and-systems-fall-2011/pages/lecture-notes/*** ***AICTE-prescribed syllabus:*** https://www.aicte-india.org/sites/default/files/Model_Curriculum/AICTE%20-%20UG%20CSE.pdf ***Industry Mapping:*** *MATLAB, SCILAB, OCTAVE*	6	1. Different operations on continuous time signals using MATLAB, Convolution, Correlation, Auto-correlation. 2. Evaluation poles, zeros and construction of transfer functions using MATLAB.	**Signals and Systems,P. Ramesh Babu, R. Anandanatarajan** Chapter 3 (Pages 3.33 - 3.88) Chapter 4 (Pages 4.10 - 4.68) Chapter 5 (Pages 5.1 - 5.61)

3	**Fourier, Laplace and z-Transforms**	Evolution of Transforms: Fourier Transform, Laplace Transform , Z-transform (single sided and Double sided) . The Fourier Transform, convolution/multiplication and their effect in the frequency domain, magnitude and phase response, Fourier domain duality. The Discrete-Time Fourier Transform (DTFT) and the Discrete Fourier Transform (DFT). Parseval's Theorem. The Laplace Transform, region of convergence, poles and zeros of system, solution to differential equations and system behavior using Laplace Transformation. The z-Transform for discrete time signals and systems, region of convergence, z-domain analysis.	***International Academia:*** (MIT Open Courseware): ***https://ocw.mit.edu/courses/6-003-signals-and-systems-fall-2011/pages/lecture-notes/*** ***AICTE-prescribed syllabus:*** https://www.aicte-india.org/sites/default/files/Model_Curriculum/AICTE%20-%20UG%20CSE.pdf ***Industry Mapping:*** *MATLAB, SCILAB, OCTAVE*	12	1. Transformation of signals into time and frequency domain Using MATLAB 2. DTFT, DFT transformation using MATLAB.	**Signals and Systems,P. Ramesh Babu, R. Anandanatarajan** Chapter 6 Chapter 7 Chapter 8 (Pages 8.1 - 8.67) Chapter 10
4	**Introduction to Sampling and Reconstruction**	The Sampling Theorem and its implications- Spectra of sampled signals. Reconstruction: ideal interpolator, zero-order hold, first-order hold, and so on. Aliasing and its effects. Relation between continuous and discrete time systems.	***International Academia:*** (MIT Open Courseware): ***https://ocw.mit.edu/courses/6-003-signals-and-***	3	1. Implementation of sampling using MATLAB. 2. Quantisation implementation using MATLAB.	**Signals and Systems,P. Ramesh Babu, R. Anandanatarajan** Chapter 9

			systems-fall-2011/pages/lecture-notes/ ***AICTE-prescribed syllabus:*** https://www.aicte-india.org/sites/default/files/Model_Curriculum/AICTE%20-%20UG%20CSE.pdf ***Industry Mapping:*** *MATLAB, SCILAB, OCTAVE*		3. Reconstruction of signals using MATLAB.	

Textbooks:

1. P.Ramesh Babu, R. Anandanatarajan, Signals and System.
2. A. V. Oppenheim, A. S. Willsky and S. H. Nawab, "Signals and systems", Prentice Hall India, 1997.
3. S. Haykin and B. V. Veen, "Signals and Systems", John Wiley and Sons, 2007.

Reference books:

1. J. G. Proakis and D. G. Manolakis, "Digital Signal Processing: Principles, Algorithms, and Applications", Pearson, 2006.
2. H. P. Hsu, "Signals and systems", Schaum's series, McGraw Hill Education, 2010.
3. A. V. Oppenheim and R. W. Schafer, "Discrete-Time Signal Processing", Prentice Hall, 2009.
4. M. J. Robert "Fundamentals of Signals and Systems", McGraw Hill Education, 2007.
5. B. P. Lathi, "Linear Systems and Signals", Oxford University Press, 2009.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	1	1	2	1	1	-	2	1	-	3
CO2	3	3	1	1	2	1	1	-	2	1	-	3
CO3	3	3	1	1	2	1	1	-	2	2	-	2
CO4	3	3	2	1	2	1	1	-	2	1	-	3

3: Strong correlation

2: Medium correlation

1: Weak correlation

University of Engineering & Management, Kolkata
University of Engineering & Management, Jaipur
Institute of Engineering & Management, Kolkata

B.Tech in Computer Science and Engineering

DETAILED SYLLABUS

Course Code- PCCCS501
Course Title – Database Management Systems
Credit – 3
Category – Professional Core Course
Semester – V
L:T:P:S – 3:0:0:0
Pre-requisite – Fundamental concepts of set theory and designing

Course Outcomes:

CO1	**Students will have a proper understanding on database system and design.**
CO2	**Students will learn the concepts of database designing using logical and mathematical concepts like relational algebra and calculus which further will be extended to learning of SQL.**
CO3	**Students will gather the understanding of relation database design through the concept of normalization.**

CO4	**Students will learn the internals of DBMS through proper understanding of transaction and further the storage architecture of data for a database system.**

Study Material	Coursera	NPTEL	Linkedin Learning	Infosys Springboard

Module No.	Topic	Sub-topics	Mapping with Industry and International Academia	Lecture Hours	Corresponding Lab Assignments	Textbook Mapping
1	**Introduction to database systems and Entity Relationship Model**	Concept & Overview of DBMS, Data Models [2L] Database Languages, Database Administrator, Database Users, Three Schema architecture of DBMS [2L] E-R modelling - Basic concepts, Design Issues, Mapping Constraints[2L] Keys, Entity-Relationship Diagram [2L] Weak Entity Sets, Extended E-R features [2L]	***International Academia:*** ***https://ocw.mit.edu/courses/6-830-database-systems-fall-2010/*** ***https://ocw.mit.edu/courses/6-5830-database-systems-fall-2023/*** ***AICTE-prescribed syllabus:*** ***https://www.aicte-india.org/sites/default/files/Model_Curriculum/AICTE%20-%20UG%20CSE.pdf*** ***StarUML Downloading link*** ***https://staruml.io/download/***	9	Designing of E-R modelling using StarUML or any other standard designing software.	**Database system concepts – By Abraham Silberschatz, Henry Korth, and S. Sudarshan (6th ed.)** Chapters: 1, 2 and 7 **Fundamentals of database systems – By Ramez Elmasri, Sham Navathe. (7th ed.)** Chapters: 1, 2 and 3

			Diagram design online using Draw.io ***https://app.diagrams.net/***			
2	**Introduction to Relational Model and SQL & Integrity Constraints**	Structure of relational Databases, Relational Algebra operations, examples and exercise [2L] Relational Calculus - operations, examples and exercise [2L] Extended Relational Algebra Operations, Views, Modifications Of the Database[2L] Concept of database languages - DDL, DML, DCL[1L] Basic Structure, Set operations, Aggregate Functions, Null Values [2L] Domain Constraints, Referential Integrity Constraints, assertions, views [2L] Joins [1L] Nested Subqueries [1L] Stored procedures and triggers, Overview of Query Optimization [1L]	***International Academia:*** ***https://ocw.mit.edu/courses/6-830-database-systems-fall-2010/*** ***https://ocw.mit.edu/courses/6-5830-database-systems-fall-2023/*** ***AICTE-prescribed syllabus:*** ***https://www.aicte-india.org/sites/default/files/Model_Curriculum/AICTE%20-%20UG%20CSE.pdf*** ***Industry Mapping: ORACLE 10g*** ***https://www.oracle.com/in/database/technologies/xe-downloads.html***	14	Queries on the following – 1. Table creation. 2. Data insertion, deletion, updation in table. 3. Aggregation functions on data. 4. Concepts of keys in table. 5. Concept of Joins. 6. Subqueries & Nested subqueries. 7. PL SQL programming. 8. Stored procedures and triggers – concept. Some of the sample queries- *https://docs.google.com/document/d/1Pbg5Y*	**Database system concepts – By Abraham Silberschatz, Henry Korth, and S. Sudarshan (6th ed.)** Chapters: 3, 4 and 6 **Fundamentals of database systems – By Ramez Elmasri, Sham Navathe. (7th ed.)** Chapters: 5, 6, 7 and 8

					kwTHC11qqjjMvbNcDHxR-2TMIb1/edit?tab=t.0	
3	**Relational Database Design**	Functional Dependency, Different anamolies in designing a Database [1L] Armstrong axioms, closure of attribute set[1L] Equivalence of functional dependency [1L] Canonical Cover [1L] Keys, types of keys, finding no. of candidate keys [2L] Normalization using funtional dependencies – 1NF, 2NF, 3NF, BCNF, multivalued dependencies - concept of 4NF, 5NF [2L] Decomposition using normal forms [1L] Losseless or Lossy decomposition [1L]	***International Academia:*** ***https://ocw.mit.edu/courses/6-830-database-systems-fall-2010/*** ***https://ocw.mit.edu/courses/6-5830-database-systems-fall-2023/*** ***AICTE-prescribed syllabus:*** ***https://www.aicte-india.org/sites/default/files/Model_Curriculum/AICTE%20-%20UG%20CSE.pdf*** ***Industry Mapping: ORACLE 10g*** ***https://www.oracle.com/in/database/technologies/xe-downloads.html***	10		**Database system concepts – By Abraham Silberschatz, Henry Korth, and S. Sudarshan (6th ed.)** Chapter: 8 **Fundamentals of database systems – By Ramez Elmasri, Sham Navathe. (7th ed.)** Chapters:14 and 15
4	**Internals of RDBMS And File Organization & Index Structures**	Concept of transactions and schedules, ACID properties [2L] Transaction processing, Concurrency control – conflict and view serializability [2L]	***International Academia:*** ***https://ocw.mit.edu/courses/6-830-database-systems-fall-2010/***	12		**Database system concepts – By Abraham Silberschatz, Henry Korth, and S.**

		Recovery Management: transaction model properties, state serializability, lock base protocols, two phase locking [2L] File & Record Concept, Placing file records on Disk, Fixed and Variable sized Records [2L] Types of Single-Level Index (primary, secondary, clustering), Multilevel Indexes[2L] Dynamic Multilevel Indexes using B tree and B+ tree [2L]	***https://ocw.mit.edu/courses/6-5830-database-systems-fall-2023/*** ***AICTE-prescribed syllabus:*** ***https://www.aicte-india.org/sites/default/files/Model_Curriculum/AICTE%20-%20UG%20CSE.pdf*** ***Industry Mapping: ORACLE 10g*** ***https://www.oracle.com/in/database/technologies/xe-downloads.html***			**Sudarshan (6th ed.)** Chapters: 11, 14, 15 and 16 **Fundamentals of database systems – By Ramez Elmasri, Sham Navathe. (7th ed.)** Chapters: 16, 17, 20 and 21

Textbooks:

1. Database system concepts – By Abraham Silberschatz, Henry Korth, and S. Sudarshan (6th ed.), McGraw-Hill.
2. Fundamentals of database systems – By Ramez Elmasri, Sham Navathe. (7th ed.), Pearson.

Reference books:

1. Database Management Systems, by Raghu Ramakrishnan, WCB/McGraw-Hill.
2. Database Management System (DBMS): A Practical Approach, by Chopra Rajiv, S. Chand Publishing.

Experiential Learning Framework (PCCCS501) Workshops, Seminars, Bootcamps, & Hackathons

This document outlines a structured framework for integrating hands-on workshops, expert seminars, intensive bootcamps, and competitive hackathon into the Database Management Systems (PCCCS501) curriculum. These initiatives are designed to close the gap between foundational theoretical computer science and active industry methodologies.

Experiential Learning Blueprint Matrix

Activity Type	Focus Module	Event Title & Theme	Core Syllabus Concepts	Deliverable / Outcome	Duration
Hands-on Workshop	Module 1: Intro & ER	Conceptualizing Digital Systems: Enterprise ERD Design	DBMS Overview, ER Modeling, Weak Entities, Mapping Constraints	Production-ready ER diagrams designed using StarUML/Draw.io for microservice system backends	1 Day (4 Hours)
Expert Seminar	Module 4: Internals	Under the Hood: Storage Architecture & Index Scaling in Modern RDBMS	File/Record Concepts, Disk Storage, Multilevel Indexes, B/B+ Trees	Interactive tracing of tree node splits, mergers, and deep dive on indexing lookups within active engines.	Half Day (2 Hours)
Intensive Bootcamp	Module 2: SQL & Rel.	Mastering Relational Query Languages & Backend Stored Procedures	SQL, Complex Joins, Nested Subqueries, Views, Triggers, PL/SQL	Rigorous code-along sandbox labs constructing enterprise schemas, optimization hints, and live query tuning.	2 Days (Weekend)
Mini Hackathon	Module 3: Rel. Design	Normalization Sprint: Fixing the Legacy Monolith	Anomalies, Functional Dependencies, Closures, 1NF to	24-hour team challenge to refactor highly redundant,	24 Hours (Continuous)

			BCNF, 4NF/5NF	intentionally unoptimized legacy schemas into optimized formats	

Strategic Implementation Milestones

- Milestone 1 (Week 3-4): Run the Enterprise ERD Workshop directly following the initial lectures on entity sets. This provides foundational clarity prior to transitioning to relational mapping math.
- Milestone 2 (Week 7-8): Execute the Intensive SQL & PL/SQL Bootcamp midway through the term. This acts as an aggressive checkpoint that dramatically accelerates students' performance in their concurrent laboratory sessions.
- Milestone 3 (Week 11): Leverage the Normalization Sprint Hackathon as a direct continuous evaluation metric or credit-earning project to maximize critical design engagement.
- Milestone 4 (Week 14): Conclude the semester with the Storage Architecture Seminar to anchor the theoretical internal behaviors of active production systems.

University of Engineering & Management, Kolkata
University of Engineering & Management, Jaipur
Institute of Engineering & Management, Kolkata

B.Tech in Computer Science and Engineering

DETAILED SYLLABUS

Course Code- PCCCS502
Course Title –Theory of Computations
Credit – 3
Category – Professional Core Course
Semester – V
L:T:P:S – 3:0:0:0
Pre-requisite – Elementary discrete mathematics includes the notion of set, function, relation, product, partial order, equivalence relation, graph & tree. They should have a thorough understanding of the principle of mathematical induction.

Course Outcomes:

CO1	**After studying Finite Automata, student will be able to define a system and recognize the behavior of a system. They will be able to minimize a system and compare different systems.**
CO2	**After studying regular language and grammar student will convert Finite Automata to regular expression. Students will be able to check equivalence between regular linear grammar and FA.**
CO3	**After studying CFG and PDA Students will be able to minimize context free grammar. Student will be able to check equivalence of CFL equivalence between regular linear grammar and FA.**

CO4	**After studying turing machine Students will be able to design Turing machine.**

Study Material	Coursera	NPTEL	Linkedin Learning	Infosys Springboard

Module No.	Topic	Sub-topics	Mapping with Industry and International Academia	Lecture Hours	Corresponding Lab Assignments	Textbook Mapping
1	**Finite Automata**	Introduction to concepts of alphabet, language, production rules, grammar and automaton, finite state model, introduction to the concept of Chomosky Classification of Grammar, language generation from production rules and vice-versa; Concept of DFA and its problems, concept of NFA and its problems. NFA to DFA conversion, Construction of DFA & NFA for any given string and vice versa, Minimization of FA and equivalence of two FA, Mealy & moore machine	***International Academia:*** ***https://ocw.mit.edu/courses/18-404j-theory-of-computation-fall-2020/pages/syllabus/*** ***AICTE-prescribed syllabus:*** ***https://www.aicte-india.org/sites/default/files/Model_Curriculum/AICTE%20-%20UG%20CSE.pdf*** ***Industry Mapping*** *JFLAP, VAS, TAGS and SimStudio*	9	1. Design a Finite State Machine (FSM) that accepts all strings over input symbols {0, 1} having three consecutive 1's as a substring. 2. Design a Finite State Machine (FSM) that accepts all strings over input symbols {0, 1} which are divisible by 3.	**Text Book 2:** Chapters: 1 and 2; Appendix A

		and their problems. Limitations of FSM.			3. Design a Finite State Machine (FSM) that accepts all decimal string which are divisible by 3.	
2	**Regular Languages and Regular Grammars**	Regular language and regular expressions, identity rules. Arden's theorem state and prove, Construction of NFA from regular expression, Conversion of NFA with null moves to without null moves, closure properties, pumping lemma and its applications, proof of pumping lemma.	***International Academia:*** ***https://ocw.mit.edu/courses/18-404j-theory-of-computation-fall-2020/pages/syllabus/*** ***AICTE-prescribed syllabus:*** ***https://www.aicte-india.org/sites/default/files/Model_Curriculum/AICTE%20-%20UG%20CSE.pdf*** ***Industry Mapping*** *JFLAP, VAS, TAGS and SimStudio*	9		**Text Book 2:** Chapters: 3 and 4;
3	**Context free Language s and machine models.**	Introduction to Context Free Grammer, Derivation trees, sentential forms. Right most and leftmost derivation of strings, concepts of ambiguity. Minimization of CFG, Chomsky normal form, Greibach normal	***International Academia:*** ***https://ocw.mit.edu/courses/18-404j-theory-of-computation-fall-2020/pages/syllabus/***	10	1. Design a Push Down Automat a (PDA) that accepts all string having equal number of 0's and 1's	**Text Book 1:** Chapters: 5 and 6;

		form, Pumping Lemma for Context Free Languages, Enumeration of properties of CFL (proofs included). Closure property of CFL, Ogden's lemma & its applications, Push Down Automata: Push down automata, definition and description, Acceptance of CFL, Acceptance by final state and acceptance by empty state and its equivalence, Equivalence of CFL and PDA, interconversion, DCFL and DPDA.	***AICTE-prescribed syllabus:*** ***https://www.aicte-india.org/sites/default/files/Model_Curriculum/AICTE%20-%20UG%20CSE.pdf*** ***Industry Mapping*** *JFLAP, VAS, TAGS and SimStudio*		over input symbol {0, 1} for a language 0n1n where n >= 1. 2. Design a Program to create PDA machine that accept the well-formed parenthesis.	
4	**Turing machines and Computability**	Turing Machine: Turing Machine, definition, model, Design of TM, Computable functions, Church's hypothesis, counter machine, Types of Turing machines (proofs not required), Universal Turing Machine, Halting problem, P, NP. Recursively enumerable (r.e.) and recursive languages. Existence of non-r.e. languages. Notion of undecidable problems. Universal language and	***International Academia:*** ***https://ocw.mit.edu/courses/18-404j-theory-of-computation-fall-2020/pages/syllabus/*** ***AICTE-prescribed syllabus:*** ***https://www.aicte-india.org/sites/default/files/Model_Curriculum/AICTE%20-%20UG%20CSE.pdf*** ***Industry Mapping***	8	1. Design a Turing Machine that calculate 2's complem ent of given binary string. 2. Design a Turing Machine which will incremen t the given binary number by 1.	**Text Book 3:** Chapters: 3 and 4;

		universal TM. Separation of recursive and r.e. classes.	*JFLAP, VAS, TAGS and SimStudio*			

Textbooks:

1. Introduction to Automata, Theory, Languages and Computation. Third Edition. John Hopcroft, Rajeev Motwani, Jeffrey D. Ullmann, Pearson Publications (Low-cost Indian edition available).
2. Peter Linz, An Introduction to Formal Languages and Automata, Narosa Pub. House, 2011
3. Introduction to the Theory of Computation, 3rd edition. Michael Sipser, Cengage Publications (Lowcost Indian edition available).

Reference books:

1. Automata and Computability, Dexter C. Kozen. Part of the Undergraduate Texts in Computer Science book series (UTCS) Springer.
2. Elements of the Theory of Computation, 2nd edition. Harry Lewis, Christos Papadimitriou, Prentice
3. Dr. R.B.Patel, Theory of Computation, Khanna Publishing House

University of Engineering & Management, Kolkata
University of Engineering & Management, Jaipur
Institute of Engineering & Management, Kolkata

B.Tech in Computer Science and Engineering

DETAILED SYLLABUS

Course Code- PCCCS503
Course Title –Operating Systems
Credit – 3
Category – Professional Core Course
Semester – V
L:T:P:S – 3:0:0:0
Pre-requisite – Basic knowledge of Data Structures and Computer Organization.

Course Outcomes:

CO1	**Explain the role, functionality, layering, abstractions, interfaces and major components of an operating system.**
CO2	**Describe process abstraction, system calls, process execution, scheduling and process-management mechanisms.**
CO3	**Analyse memory-management mechanisms, concurrency problems and synchronization-based solutions.**
CO4	**Explain file-system abstractions, file-system interfaces, persistent storage and file-system design principles.**

Study Material	Coursera	NPTEL	LinkedInLearning	Infosys Springboard

Module No.	Topic	Sub-topics	Mapping with Industry and International Academia	Lecture Hours	Textbook Mapping
1	**Introduction to Operating Systems**	Application requirements; systems stack and role of the operating system; resources, abstractions and interfaces; overview of operating-system components; examples and characteristics of different types of operating systems such as real-time, desktop and mobile operating systems; operating systems and operating-system distributions.	***International Academia:*** **https://online.stanford.edu/courses/cs111-operating-systems-principles** ***AICTE-prescribed syllabus:*** **AICTE Model Curriculum for UG Degree Course in CSE (PCC CS-403)** ***Industry Mapping*** *Linux/UNIX, Android, real-time operating systems.*	2	**Text Book 1: Ch. 2; Text Book 2: Ch. 1-2; Text Book 3: Ch. 1**
2	**Computer Organization and Architecture Refresher**	Basic organization of hardware components; role of the operating system relative to hardware functionality, with examples related to the von Neumann architecture.	***International Academia:*** **https://online.stanford.edu/courses/cs111-operating-systems-principles** ***AICTE-prescribed syllabus:*** **AICTE Model Curriculum for UG Degree Course in CSE (PCC CS-403)** ***Industry Mapping*** *CPU privilege modes, interrupts and system-call interface.*	1	**Text Book 2: Ch. 1; Text Book 3: Ch. 1**
3	**Processes**	Process abstraction: program versus process; Process Control Block (PCB); design of system calls, their invocation and basic operating-system handling; process-control system calls such as fork, wait, exec, getpid, getppid and variants; limited direct execution model.	***International Academia:*** **https://online.stanford.edu/courses/cs111-operating-systems-principles** ***AICTE-prescribed syllabus:*** **AICTE Model Curriculum for UG Degree Course in CSE (PCC CS-403)** ***Industry Mapping*** *POSIX process API, Linux process model.*	5	**Text Book 1: Ch. 4-6; Text Book 2: Ch. 6-7; Text Book 3: Ch. 7-8**
4	**Memory Management**	Address bus and memory access; memory view of a process: heap, stack,	***International Academia:*** **https://online.stanford.edu/courses/cs111-**	9	**Text Book 1: Ch. 13-**

Module No.	Topic	Sub-topics	Mapping with Industry and International Academia	Lecture Hours	Textbook Mapping
		code and data; process memory requirements; address-space abstraction using virtual memory and related system calls; address translation mechanisms: static mapping, segmentation and paging; page faults, page sharing, read/write permissions, swapping, process versus OS memory; motivation and mechanisms for memory bookkeeping and management; case studies of malloc and the role of the OS in transforming a program into a process.	**operating-systems-principles** ***AICTE-prescribed syllabus:*** **AICTE Model Curriculum for UG Degree Course in CSE (PCC CS-403)** ***Industry Mapping*** *virtual memory, paging, memory protection, Linux process address space.*		**20; Text Book 2: Ch. 9**
5	**Process Management**	Process life cycle from source code to execution; operating-system mode of execution; limited direct execution, interrupts and system calls; process context-switch mechanism and PCB state; scheduling policies, scheduling metrics and goals; examples involving interactive, real-time and priority scheduling.	***International Academia:*** **https://online.stanford.edu/courses/cs111-operating-systems-principles** ***AICTE-prescribed syllabus:*** **AICTE Model Curriculum for UG Degree Course in CSE (PCC CS-403)** ***Industry Mapping*** *Linux schedulers, process priorities and context switching.*	4	**Text Book 1: Ch. 7-8; Text Book 2: Ch. 8**
6	**Concurrency and Synchronization**	Motivation and application, process and OS use cases; introduction to threads and the pthread API; synchronization primitives; limitations of software-only solutions; atomic instructions, test-and-set, spinlocks, mutexes, condition variables and semaphores; pthread synchronization API; case studies: producer-consumer, reader-writers and barriers; issues with concurrency	***International Academia:*** **https://online.stanford.edu/courses/cs111-operating-systems-principles** ***AICTE-prescribed syllabus:*** **AICTE Model Curriculum for UG Degree Course in CSE (PCC CS-403)** ***Industry Mapping*** *POSIX threads, mutexes, semaphores and*	9	**Text Book 1: Ch. 26-32; Text Book 3: Ch. 11-12**

Module No.	Topic	Sub-topics	Mapping with Industry and International Academia	Lecture Hours	Textbook Mapping
		including race conditions, deadlocks and order violations.	*concurrent systems.*		
7	**File Systems**	Persistence and the file abstraction; hardware view of hard-disk architecture and interfacing; process view and system calls for file handling; roles and responsibilities of a file system; file-system design details: file and file-system metadata, directory structure and caching optimizations; file-system case study such as the UNIX file system.	***International Academia:*** **https://online.stanford.edu/courses/cs111-operating-systems-principles** ***AICTE-prescribed syllabus:*** **AICTE Model Curriculum for UG Degree Course in CSE (PCC CS-403)** ***Industry Mapping*** *POSIX file API, UNIX/Linux file systems.*	6	**Text Book 1: Ch. 36-40; Text Book 2: Ch. 4-5; Text Book 3: Ch. 3**

Textbooks:

1. Operating Systems: Three Easy Pieces, Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau, Arpaci-Dusseau Books, LLC.
2. Operating System Concepts, Abraham Silberschatz, Peter B. Galvin and Greg Gagne, Wiley.
3. Operating Systems: Internals and Design Principles, William Stallings, Pearson.

Reference books / Online Resources:

1. Modern Operating Systems, Andrew S. Tanenbaum and Herbert Bos, Pearson.
2. Operating Systems: Principles and Practice, Thomas Anderson and Michael Dahlin, Recursive Books.
3. AICTE Model Curriculum for UG Degree Course in Computer Science and Engineering (2022), PCC CS-403 Operating Systems.
4. Stanford CS111: Operating Systems Principles; MIT 6.S081: Operating System Engineering.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	1							
CO2	3	3	2	2	1							
CO3	3	3	2	2	1							
CO4	3	2	2	1	2							

University of Engineering & Management, Kolkata
University of Engineering & Management, Jaipur
Institute of Engineering & Management, Kolkata

B.Tech in Computer Science and Engineering

DETAILED SYLLABUS

Course Code- PCCCS504
Course Title –Software Engineering and Project Management
Credit – 3
Category – Professional Core Course
Semester – V
L:T:P:S – 3:0:3:0
Pre-requisite – Basic knowledge of Computer Science, Proficiency in languages like C, C++, Python, Java.

Course Outcomes:

CO1	**Students will be able to remember the given project in various phases of a life cycle as well as various cost-benefit analyses.**
CO2	**Students will be able to understand and specify software requirements and understand design concepts, like Decision Trees, Decision tables, DFD, Structure Charts, and UML diagrams, and then realize that design practically, using an appropriate software engineering methodology.**
CO3	**Students will be able to analyze the process of writing the code from the design, effectively apply relevant standards, and perform testing and quality assurance.**
CO4	**Students will understand the end-to-end project management, risk management concepts along with various testing plans and quality aspects.**

Study Material	Coursera	NPTEL	LinkedInLearning	Infosys Springboard

Module No.	Topic	Sub-topics	Mapping with Industry and International Academia	Lecture Hours	Corresponding Lab Assignments	Textbook Mapping

1	**Introduction and Software Process Models**	Software, Software Engineering, Myths, Software Process, Work Products, Importance of Software Engineering, Standard for Software Process, Waterfall Model, Prototyping Model, Iterative Enhancement Model, Spiral Model, RAD model, 4th Generation models, Formal Methods, Agile Development	***International Academia:*** *https://ocw.mit.edu/courses/16-355j-software-engineering-concepts-fall-2005/* ***AICTE-prescribed syllabus:*** https://www.aicte-india.org/sites/default/files/Model_Curriculum/AICTE%20-%20UG%20CSE.pdf ***Industry Mapping:*** IEEE SRS standard, Rational Rose, Reqview, Jira software,Axosoft	6	**1.Software Process Simulation**: Scenario: You are leading a team to develop a mobile banking app. Choose a software process model (Waterfall, Agile, Spiral, etc.) and simulate its application.	**Text book 1** Chapters: 1, 2, 3 **Text book 2** Chapters 1, 2 **Text book 3** Chapters 1, 2
2	**Requirement Engineering and Software Project Management**	Requirement Engineering and Software Project Management Software Requirements, Types of Requirements, Requirement Engineering Cycle, Requirements Specification document, Characteristics of Requirements, Requirement verification and validation, Role of Management in Software Development,	***International Academia:*** *https://ocw.mit.edu/courses/16-355j-software-engineering-concepts-fall-2005/* ***AICTE-prescribed syllabus:*** https://www.aicte-india.org/sites/default/files/Model_Curriculum/AICTE%20-%20UG%20CSE.pdf ***Industry Mapping:*** MS project, ProjectLibre, FunctionPointmodeler	8	2. **Requirements Gathering and Analysis:** Scenario: A local bakery wants to digitize its ordering system. Conduct a requirements gathering session with the bakery owner (role-play this with a colleague). Prepare functional and non-functional requirement lists.	**Text book 1** Chapters: 5, 6 **Text book 2** Chapters 4 **Text book 3** Chapters 3

		Project Estimation Techniques, Staffing, Scheduling, Earned Value Analysis, Software Risks, Software Configuration Management, Software Process and Project metrics.				
3	**Software Design and Coding**	Process, Data and Behavioural Modelling, Design Concepts, Modularity, Architectural design, Coupling and Cohesion, Top-down and bottom-up design, Object-oriented Analysis, Function oriented and Object-Oriented Design approach, Software Design Document, Coding styles and documentation	***International Academia:*** *https://ocw.mit.edu/courses/16-355j-software-engineering-concepts-fall-2005/* ***AICTE-prescribed syllabus:*** https://www.aicte-india.org/sites/default/files/Model_Curriculum/AICTE%20-%20UG%20CSE.pdf ***Industry Mapping:*** IEEE SDD document. Smart draw, Visual Paradigm/Microsoft Visio/MS Project/Umbrello/Rational Rose.	8	**3. Prototyping a User Interface:** Scenario: Design a prototype for a user interface for a smart home control system. Use a prototyping tool (Figma, Balsamiq, Adobe XD) to create a visual representation of the interface.	**Text book 2** Chapters: 5, 6 and 7
4	**Testing and Software Quality**	Testing principles, testing strategies, Black-box and White-box Testing Techniques, Levels of testing -unit, integration, system, regression, Test Plan, Test Cases Specification,	***International Academia:*** *https://ocw.mit.edu/courses/16-355j-software-engineering-concepts-fall-2005/* ***AICTE-prescribed syllabus:*** https://www.aicte-india.org/sites/default/files/Model_Curriculum/AICTE%20-%20UG%20CSE.pdf	8	**4. Testing a Software Module:** Scenario: Write test cases for a function that validates user input in a registration form.	**Text book 1** Chapters: 17 and 18 **Text book 2** Chapters: 10 and 11

		Software debugging, Software Maintenance, Software Quality Assurance (SQA), SQA tasks, Software amplification and removal, Formal Technical Reviews, Software Quality Factors, ISO 9126, SEI CMM, CMMI, Software Reliability. Software Availability.	***Industry Mapping:*** Eclipse, Bugzilla, MantisBT, Jira Software		Include different types of tests (e.g., boundary value analysis, equivalence partitioning). Execute the tests and document the results. Discuss the importance of testing in ensuring software quality.	**Text book 3** Chapter: 10
5	**Computer Aided Software Engineering and Advanced Topics**	Computer Aided Software Engineering (CASE) and its Scope, CASE support in Software Life Cycle, Architecture of CASE Environment, Upper CASE and Lower CASE, Exposure to CASE tools. Software Process Improvement, Component Based Software Engineering, Web Engineering, Reverse Engineering, Software	***International Academia:*** *https://ocw.mit.edu/courses/16-355j-software-engineering-concepts-fall-2005/* ***AICTE-prescribed syllabus:*** https://www.aicte-india.org/sites/default/files/Model_Curriculum/AICTE%20-%20UG%20CSE.pdf ***Industry Mapping:*** IEEE SDD document. Smart draw, Visual Paradigm/Microsoft Visio/MS Project/Umbrello/Rational Rose.	6	**5. CASE Tool-Based Software Design** **Scenario:** A software development company is modernizing its existing Library Management System. The team needs to document the system, redesign certain modules, and improve maintainability using CASE tools.	**Text book 2** Chapters 12

		Engineering challenges of Big Data, Mobile Applications.				

Textbooks:

1. Pressman, Software Engineering: A practitioner's approach– (TMH)
2. Rajib Mall, Fundamentals of Software Engineering- (PHI)
3. Pankaj Jalote, An Integrated Approach to Software Engineering- (Wiley-India)

Reference books:

1. Saikat Dutt, S. Chandramouli, Software Engineering - Pearson
2. Agarwal and Agarwal, Software Engineering – (PHI)
3. Sommerville, Software Engineering – Pearson
4. Martin L. Shooman, Software Engineering – TMH

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	2	2				1			2
CO2	3	3	3	2	2							2
CO3	3	3		3								2
CO4	3	3		3					1			2

University of Engineering & Management, Kolkata
University of Engineering & Management, Jaipur
Institute of Engineering & Management, Kolkata

B.Tech in Computer Science and Engineering

DETAILED SYLLABUS

Course Code- PCCCS582
Course Title –Neural Network & Deep Learning
Credit – 2
Category – Professional Core Course
Semester – V
L:T:P:S – 0:0:0:4
Pre-requisite – Basic knowledge of Machine learning.

Course Outcomes:

CO1	**Understand the foundational concepts of neural networks, perceptron, and the architecture of deep neural models.**
CO2	**Analyze and implement deep learning training algorithms including backpropagation, optimization, and regularization techniques.**
CO3	**Design and evaluate convolutional neural networks (CNNs) for visual data processing, including object detection and segmentation.**

CO4	**Apply sequence-based models such as RNN, LSTM, GRU, and bidirectional architectures for tasks involving sequential data and language modeling.**

Study Material	Coursera	NPTEL	LinkedInLearning	Infosys Springboard

Module No.	Topic	Sub-topics	Mapping with Industry and International Academia	Lecture Hours	Corresponding Lab Assignments	Textbook Mapping
1	**Introduction to NN and Deep Learning**	Basics of artificial neural networks (ANN): Understanding biological neurons and artificial neurons; Perceptron, XOR problem, Computational models of neurons, Structure of neural networks, Functional units of ANN for pattern recognition tasks. multi-layered perceptron, Types of activation functions; Architectures of neural network; Learning process in ANN.	IIT Mandi syllabus (CS671) ***International Academia:*** ***https://www.coursera.org/specializations/deep-learning*** *international curriculum of Stanford CS231n* ***Industry Mapping:*** *TensorFlow, Keras, PyTorch*	5	1. XOR: Train ANN from scratch, plot decision boundary & loss. 2. MNIST: Use Keras/PyTorch with ≥3 layers, test activations. 3. Optimizers: Compare SGD, Adam, RMSprop on Fashion-MNIST. 4. Regularization: Use Dropout & BatchNorm, compare results. 5. Visualization: Plot computational graph using	**Textbook: Deep Learning by Ian Goodfellow, Yoshua Bengio, Aaron Courville** Chapters: 5, 6, 7 and 8 **Reference book: Neural Networks and Learning Machines by Simon Haykin**

					TensorBoard/torchviz. Assignment questions: **Explain** the role of different activation functions (ReLU, Sigmoid, Tanh) in neural networks. Compare their performance on a toy dataset. **Illustrate** the impact of improper weight initialization using a multi layer perceptron trained on the XOR problem. **Derive** the mathematical formulation of the backpropagation algorithm for a simple 3-layer network. **Differentiate** between Batch Gradient Descent, Mini-Batch Gradient Descent,	Chapters: 4 and 1

					and Stochastic Gradient Descent with real-life analogies and convergence behavior. **Explain** the importance of normalization and regularization in deep learning models. Provide examples of each.	
2	**Deep Learning: Principles and Training Algorithms**	Why Is Training Deep Networks Hard? Cliff, Valley, Convergence over depth of network, Local Minima, Dying Neurons, Training Deep Neural Network: Backpropagation and mathematics behind it, Weight initialization in a neural network, Batch, mini-batch, and stochastic gradient descent, Optimization algorithms, Regularization, and Normalization.	IIT Mandi syllabus (CS671) ***International Academia:*** ***https://www.coursera.org/specializations/deep-learning*** *international curriculum of MIT 6.S191* ***Industry Mapping:*** *TensorFlow, Keras, PyTorch*	5	Hands-On Questions (Brief): 1. Gradient Flow: Visualize vanishing/exploding gradients in deep MLP. 2. Init Impact: Compare training using Random, Xavier, He init. 3. Activation Check: Detect dead neurons for ReLU, Leaky ReLU, Tanh. 4. Loss Landscape: Plot optimization surface for deep model.	**Textbook: Deep Learning by Ian Goodfellow, Yoshua Bengio, Aaron Courville** Chapters: 5, 6, 7 and 8 **Reference book: Neural Networks and Deep Learning by Charu C. Agarwal**

					5. Optimizer Test: Compare SGD, Adam, RMSprop on deep network Assignment Questions: 1. Explain cliff/valley issues in deep networks. 2. What are dying neurons? How to avoid? 3. Role of local minima and saddle points. 4. Compare Xavier, He, and Random initialization methods. 5. Batch vs Mini-batch vs SGD – pros and cons.	Chapter: 4
3	**Convolutional Neural Network**	Historical Perspective and Biological Inspiration; Challenges faced by traditional ANN to deal with image data; Convolutional neural network concepts – kernel, stride, padding, pooling; Fully Connected Layers, Building a CNN;	***AICTE-prescribed syllabus:*** ***https://www.aicte-india.org/sites/default/files/Model_Curriculum/CS%20(AI&ML).pdf*** ***International Academia:***	6	Assignment Questions (Brief): 1. Explain why traditional ANN struggles with image data. 2. Describe the role of kernel, stride, and padding in CNNs.	**Textbook: Deep Learning by Ian Goodfellow, Yoshua Bengio, Aaron Courville:**

		Backpropagation as Convolution with Inverted/Transposed Filter, Popular CNN architectures – LeNet, AlexNet, GoogLeNet, ResNet, Inception network, UNET; Object detection – bounding box, YOLO, landmark detection, Transfer learning. Natural Language and Sequence Learning with TextCNN	***https://www.coursera.org/learn/convolutional-neural-networks***		3. Compare CNN architectures: LeNet, AlexNet, GoogLeNet, ResNet. 4. What is transposed convolution? Why is it needed? 5. Explain object detection using bounding boxes and YOLO. **Hands-On Questions (Brief):** 1. CNN Basics: Build a CNN on CIFAR-10. Vary kernel/padding and observe accuracy. 2. Architecture Test: Implement LeNet and ResNet on same dataset; compare performance. 3. YOLO Demo: Use pretrained YOLO for real time object	Chapter: 9 **Reference books:** **Deep Learning by Amit Kumar Das, Saptarsi Goswami, Pabitra Mitra and Amlan Chakrabarti** Chapter: 6 **Neural Networks and Deep Learning by Charu C. Agarwal** Chapter: 9

					detection on webcam/video. 4. UNET Task: Apply UNET for basic image segmentation. 5. TextCNN: Use TextCNN to classify movie reviews or tweets (binary sentiment).	
4	**Sequence-Based Models**	Introduction to sequence data; Recurrent neural network; Vanishing Gradient Problem and RNN; Long Short-term Memory (LSTM); Gated Recurrent Units (GRU); Bi-directional Models;	***AICTE-prescribed syllabus:*** ***https://www.aicte-india.org/sites/default/files/Model_Curriculum/CS%20(AI&ML).pdf*** ***International Academia:*** ***https://www.coursera.org/learn/nlp-sequence-models***	5	Assignment Questions (Brief): 1. Explain vanishing gradient problem in RNNs. 2. Differentiate between RNN, LSTM, and GRU. 3. Describe how bidirectional models improve learning. 4. Discuss the role of memory cells in LSTM. 5. Compare the computational complexity of LSTM vs GRU. **Hands-On Questions:**	**Textbook: Deep Learning by Ian Goodfellow, Yoshua Bengio, Aaron Courville** Chapter: 10 **Reference book: Deep Learning by Amit Kumar Das, Saptarsi Goswami, Pabitra Mitra and**

					1. Char-level RNN: Train a basic RNN to generate character sequences from text. 2. LSTM Language Model: Build an LSTM for next word prediction on a text corpus. 3. GRU vs LSTM: Train both on same dataset; compare accuracy and training time. 4. Bidirectional LSTM: Apply Bi-LSTM for sentiment analysis on IMDB dataset. 5. Gradient Check: Visualize gradient flow in deep RNN to demonstrate vanishing gradients.	**Amlan Chakrabarti** Chapter: 8
5	**Introduction to Generative Models**	The Concept of Generative Modelling, Benefits and Challenges, Generative Adversarial Networks	***AICTE-prescribed syllabus:*** ***https://www.aicte-india.org/sites/default/files/***	6	**Assignment Questions (Brief):**	**Textbook: Generative Deep Learning:**

		(GANs): Architecture, Training Process, Applications (Deepfakes, Style Transfer) - Variational Autoencoders (VAEs): Architecture, Training Process, Applications (Anomaly Detection, Data Augmentation)	***Model_Curriculum/CS%20(AI&ML).pdf*** ***International Academia:*** ***https://www.coursera.org/learn/generative-ai-introduction-and-applications***		1. Differentiate between generative and discriminative models. 2. Explain the architecture and loss functions in GANs. 3. What challenges arise when training GANs? 4. Discuss applications of VAEs in real-world tasks. **Hands-On Questions (Brief):** 1. GAN Training: Use PyTorch or Keras to build a GAN for digit generation (MNIST). 2. VAE Demo: Implement a VAE for anomaly detection on tabular data. 3. Style Transfer: Use a pre-trained GAN to perform	**Teaching Machines to Paint, Write, Compose, and Play by David Foster** Chapter 1, 3, 4, 5, 8 and 9 - Handouts

					style transfer on images. 4. Data Augmentation: Use VAE to generate synthetic samples for training.	

Textbooks:

1. Deep Learning by Ian Goodfellow, Yoshua Bengio, Aaron Courville
2. Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play by David Foster

Reference books:

1. Neural Networks and Learning Machines by Simon Haykin
2. Neural Networks and Deep Learning by Charu C. Agarwal
3. Deep Learning by Amit Kumar Das, Saptarsi Goswami, Pabitra Mitra and Amlan Chakrabarti

University of Engineering & Management, Kolkata
University of Engineering & Management, Jaipur
Institute of Engineering & Management, Kolkata

B.Tech in Computer Science and Engineering

DETAILED SYLLABUS

Course Code- PCCCS581
Course Title –Quantum Computing
Credit – 2
Category – Professional Core Course (Sessional)
Semester – V
L:T:P:S – 0:0:0:4
Pre-requisite – Linear Algebra

Module No.	Topic	Sub-topics	Lecture Hours
1	**Introduction to Quantum Computing**	**1.1 Motivation and Context** • Motivation for studying Quantum Computing • Major players in the industry (IBM, Microsoft, Rigetti, D-Wave, etc.)	8

		• Origin and historical development of Quantum Computing **1.2 Elementary Quantum Mechanics and Linear Algebra for Quantum Computing** • Qubits and multi-qubit states; Bra-ket (Dirac) notation • Quantum states in Hilbert space • The Bloch sphere representation • Quantum superposition • Quantum entanglement	
2	**Quantum Correlations and Mathematical Foundations**	**2.1 Mathematical Foundation for Quantum Computing** • Basis vectors and orthogonality • Inner product and Hilbert spaces • Matrices and tensors • Unitary operators and projectors • Dirac notation (formal treatment) • Eigenvalues and eigenvectors **2.2 Quantum Correlations** • Bell inequalities and entanglement • Quantum teleportation	8
3	**Quantum Cryptography and Building Blocks for a Quantum Program**	**3.1 Quantum Cryptography** • Quantum Key Distribution (QKD) **3.2 Architecture of a Quantum Computing Platform** • Details of the qubit system of information representation • Bloch sphere (applied/architectural view) • Multi-qubit states	8

		• Quantum superposition of qubits (valid and invalid superposition) • Quantum entanglement (applied treatment) • Useful states from an algorithmic perspective, e.g., Bell states **3.3 Operations on Qubits** • Measuring and transforming qubits using gates • Quantum logic gates and circuits: Pauli, Hadamard, phase shift, controlled gates, swap, etc. **3.4 Programming Model for a Quantum Computing Program** • Steps performed on the classical computer • Steps performed on the quantum computer • Moving data between bits and qubits	
4	**Quantum Gates and Algorithms**	**4.1 Universal Gate Sets and Circuits** • Universal set of gates • Quantum circuits **4.2 Basic Techniques Exploited by Quantum Algorithms** • Amplitude amplification • Quantum Fourier Transform • Phase kick-back • Quantum phase estimation **4.3 Major Quantum Algorithms** • Deutsch's algorithm • Deutsch-Jozsa algorithm • Shor's algorithm (factoring) • Grover's search algorithm	8

5	**Programming a Quantum Computer**	**5.1 Hands-on Qiskit** • Performing basic operations using Qiskit • Coding a quantum computer using a simulator to carry out basic quantum measurement and state analysis • Construction of qubits and different types of quantum gates	8

Textbooks:

1. Michael A. Nielsen and Issac L. Chuang, "Quantum Computation and Information, Cambridge (2002).

Reference books:

1. Mikio Nakahara and Tetsuo Ohmi, "Quantum Computing", CRC Press (2008).
2. N. David Mermin, "Quantum Computer Science", Cambridge (2007)

University of Engineering & Management, Kolkata
University of Engineering & Management, Jaipur
Institute of Engineering & Management, Kolkata

B.Tech in Computer Science and Engineering

DETAILED SYLLABUS

Course Code- MCC571
Course Title –Constitution of India
Credit – 1
Category – Mandatory Course
Semester – V
L:T:P:S – 1:0:0:0
Pre-requisite – Some idea about professional life and society.

Course Outcomes:

CO1	Remembering and understanding the salient features of the Indian Constitution.
CO2	Analyzing the workings of Union, State and local governments.
CO3	Identifying and analyzing the function of the judiciary.
CO4	Understanding the function of local Governments and developing attitude and skills for critical analysis of social policy and development plans.

Module No.	Topic	Sub-topics	Mapping with Industry and International Academia	Lecture Hours	Textbook Mapping
1	**Introduction to Indian Constitution**	Indian Constitution: Sources and constitutional history Features: Citizenship, Preamble, Fundamental Rights and Duties Directive Principles of State Policy		2	**Textbook1:** Chapters: 1, 2, 3, 4, 6, 7, 8 and 9
2	**Union**	Union government and its administration: Structure of the Indian Union Centre- State relationship President: Role, power and position PM and Council of ministers, Cabinet and Central Secretariat Lok Sabha, Rajya Sabha State government and its administration: Governor: Role and Position, CM and Council of ministers, State Secretariat:		3	**Textbook1:** Chapters: 12,13,14,15,17,18, 19, 20, 22, 31, 32 and 33
		Organisation, Structure and Functions			

3	**Judiciary**	Supreme court: Organization of supreme court, procedure of the court, independence of the court, jurisdiction and power of supreme court High court: Organization of high court, procedure of the court, independence of the court, jurisdiction Subordinate courts: constitutional provision, structure and jurisdiction National legal services authority, Lok adalats, family courts, gram-nyayalays. Public interest litigation (PIL): meaning of PIL, features of PIL, scope of PIL, principle of PIL, guidelines for admitting PIL		2	**Textbook1:** Chapters: 26-29, 34-37
4	**Local Government**	Local Administration: District's Administration head: Role and Importance, Municipalities: Introduction, Mayor and role of Elected Representative, CEOof Municipal Corporation Pachayatiraj: Introduction, PRI: Zila Pachayat, Elected officials		3	**Textbook1:** Chapters: 28-41

		and their roles, CEO Zila Pachayat: Position and role, Block level: Organizational Hierarchy (Different departments), Village level: Role of Elected and Appointed officials, Importance of grassroot democracy			

Textbooks:

1. Indian polity, M. Laxmikanth, MC Graw Hill education, 5th Edition.

Reference books:

1. D D Basu, “IntroductiontotheconstitutionofIndia”,21st Edition, Lexis Nexis Books Publication ltd, India.

University of Engineering & Management, Kolkata
University of Engineering & Management, Jaipur
Institute of Engineering & Management, Kolkata

B.Tech in Computer Science and Engineering

DETAILED SYLLABUS

Course Code- ESPCS501
Course Title – Essential Studies for Professionals (CS) - V
Credit – 0.5
Category – Humanities & Social Sciences including Management course
Semester – V
L:T:P:S – 2:0:0:0
Prerequisite – Digital Electronics, Programming and Data Structures, Computer Organization & Architecture, Discrete Mathematics
Course Outcomes:

CO1	**Students would be able to understand the mathematical foundations required for solving computational problems.**
CO2	**Students would be able to analyze and design combinational and sequential digital circuits.**
CO3	**Students would be able to develop efficient algorithms using C programming and data structures.**
CO4	**Students would be able to understand computer organization concepts including instruction execution, pipelining, memory hierarchy and computer arithmetic.**

Module number	Topic	Sub-topics	Mapping with Industry and International Academia	Lecture Hours	Corresponding Lab Assignment
1	**Digital Logic:** Digital Logic Boolean algebra. Combinational and sequential circuits. Minimization. Number representations and computer arithmetic (fixed and floating point). **(Ch 1, Ch 2, Ch 3, Ch 4, Ch 5, Ch 6: M. Morris Mano, Digital Logic and Computer Design, Pearson)**	• Boolean Algebra • Logic Gates • Canonical Forms • Karnaugh Maps • Combinational Circuits • Multiplexers • Decoders • Encoders • Adders • Sequential Circuits • Flip-Flops • Registers • Counters • Number Systems • Fixed-point Arithmetic • Floating-point Arithmetic	***International Academia:*** 1. https://ocw.mit.edu/courses/6-004-computation-structures-spring-2017/ 2. https://nptel.ac.in/courses/108105113 3. https://courses.ece.cmu.edu/18240	6L	• K-map minimization • Design of combinational circuits • Flip-flop conversion • Counter design • Previous GATE problems

2	**Programming and Data Structures** Programming and Data Structures Programming in C. Recursion. Arrays, stacks. **(Ch 1 – Ch 9: Ellis Horowitz, Sartaj Sahni, Fundamentals of Data Structures, Universities Press)**	Programming in C • Variables • Functions • Recursion • Pointers • Arrays • Strings • Structures Data Structures • Stack • Queue • Linked List • Trees (Introduction) • Time Complexity • Searching • Sorting	***International Academia:*** 1. https://pll.harvard.edu/course/cs50-introduction-computer-science 2. https://web.stanford.edu/class/cs106b/ 3. https://nptel.ac.in/courses/106104128 ***AICTE-prescribed syllabus:*** https://www.aicte-india.org/sites/default/files/Model_Curriculum/AICTE%20-%20UG%20CSE.pdf	6L	• Recursive programs • Stack implementation • Queue implementation • Linked List implementation • Complexity analysis • GATE previous -year questions

3	**Computer Organization and Architecture:** Machine instructions and addressing modes. ALU, data path and control unit. Instruction pipelining. **(Ch 1, Ch 2, Ch 5, Ch 6 – Ch 8: Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Computer Organization, McGraw-Hill)**	• Instruction Set Architecture • Addressing Modes • CPU Organization • ALU • Datapath • Control Unit • Hardwired Control • Microprogrammed Control • Instruction Pipeline • Pipeline Hazards • Memory Hierarchy • Cache Basics	***International Academia:*** 1. https://ocw.mit.edu/courses/6-823-computer-system-architecture-fall-2005/ 2. https://cs61c.org/su26/ 3. https://nptel.ac.in/courses/106105163	6L	• Addressing mode exercises • Pipeline hazard problems • Cache mapping problems • Previous GATE numerical problems
4	**Discrete Mathematics:** Propositional and first order logic. Sets, relations, functions, partial orders and lattices. Groups. Graphs: connectivity, matching, coloring. Combinatorics: counting, recurrence relations, generating functions. **(Ch 1, Ch 2, Ch 6, Ch 8 – Ch 11: Kenneth H. Rosen, Discrete Mathematics and Its Applications, McGraw-Hill)**	• Propositional Logic • Predicate Logic • Sets • Relations • Functions • Partial Orders • Lattices • Groups • Graph Theory • Trees • Connectivity • Coloring • Matching • Counting • Recurrence Relations • Generating Functions	***International Academia:*** 1. https://ocw.mit.edu/courses/6-042j-mathematics-for-computer-science-fall-2010/ 2. https://online.stanford.edu/courses/cs103-mathematical-foundations-computing ***AICTE-prescribed syllabus:*** https://www.aicte-india.org/sites/default/files/Model_Curriculum/AICTE%20-%20UG%20CSE.pdf	6L	• Logic problems • Graph problems • Counting problems • Recurrence solving • Previous GATE questions

Textbooks:

1. M. Morris Mano, Digital Logic and Computer Design, Pearson.
2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Computer Organization, McGraw-Hill.
3. Ellis Horowitz, Sartaj Sahni, Fundamentals of Data Structures, Universities Press.
4. Kenneth H. Rosen, Discrete Mathematics and Its Applications, McGraw-Hill.

Reference Books:

1. R. P. Jain, Modern Digital Electronics.
2. Narasimha Karumanchi, Data Structures and Algorithms Made Easy.
3. William Stallings, Computer Organization and Architecture.
4. Susanna Epp, Discrete Mathematics with Applications.
5. GATE Computer Science Study Material (Made Easy Publications).