SCHEME OF INSTRUCTION AND DETAILED SYLLABI FOR

MINOR DEGREE

IN

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Effective from 2018 Batch onwards

Dr. B R AMBEDKAR NATIONAL INSTITUTE OF TECHNOLOGY

JALANDHAR – 144011

Phone: 0181-2690301, 02 (Ext. 2502) **Fax: 0181-2690932** **Website: www.nitj.ac.in**

ANNEXURE-II

As per the institute recommendations, following B Tech teaching scheme and course curriculum for the minor degree have been proposed for the discussion/consideration of the honorable Board of Studies (BOS) for the kind approval.

Total credit 18

Minor Elective Courses

S.No.	Course Code	Courses	L	T	P	Credit
1.	CSMI-201	Relational Data Base Management System	3	0	0	3
2.	CSMI-202	Computer Organization and Architecture	3	0	0	3
3.	CSMI-301	Operating System	3	0	0	3
4.	CSMI-302	Object-Oriented Programming	3	0	0	3
5.	CSMI-401	Software Engineering	3	0	0	3
6.	CSMI-402	Data Analytics	3	0	0	3
Total Credit			**18**	**0**	**0**	**18**

DETAILED COURSE CONTENT OF MINOR ELECTIVES

DEPARTMENT: COMPUTER SCIENCE AND ENGINEERING
COURSE CODE: CSMI-201
COURSE TITLE: RELATIONAL DATA BASE MANAGEMENT SYSTEM
COURSE DESIGNATION: REQUIRED
PRE-REQUISITES: NONE
CONTACT HOURS/CREDIT SCHEME :(L-T-P-C: 3-0-0-3)
COURSE ASSESSMENT METHOD: Two sessional exams and one end-semester exam, along with assignments, presentations and class tests which may be conducted by the course coordinator in lieu of internal assessment.

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

1. Differentiate database systems from file systems by enumerating the features provided by database systems and describe each in both function and benefit.
2. Implement a relational database into a database management system.
3. Analyze an information storage problem and derive an information model expressed in the form of an entity relation diagram and other optional analysis forms, such as a data dictionary.
4. Transform an information model into a relational database schema and to use a data definition language and/or utilities to implement the schema using a DBMS.
5. Formulate, using SQL, solutions to a broad range of query and data update problems

Course Outcomes	Program outcomes											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CSMI-201												
CO 1.	L	H	M	M		M						
CO 2.	L					M		M				
CO 3.		L	H	H	M	H	M		H		H	
CO 4.		H	M	M			H	H				H
CO 5.	H			M	H							

TOPICS COVERED

Basic Concepts: Introduction to File and Database systems- Database system structure – concepts and architecture, date models, schemas & instances, DBMS architecture & data independence, database languages & interfaces, Data Model, ER model.

Relational Models: SQL – Data definition- Queries in SQL-relational model concepts, relational model constraints, relational algebra, SQL- a relational database language: date definition in SQL, view and queries in SQL, specifying constraints and indexes in SQL; relational database management systems-Updates, Views, Integrity and Security, Relational Database design, Functional dependences and Normalization for Relational Databases, normal forms based on primary keys, (1NF, 2NF, 3NF & BCNF), lossless join and dependency preserving decomposition, converting ER-diagrams into relations.

Data Storage and query Processing: Record storage and Primary file organization- Secondary storage Devices, Operations on Files, Heap File, Sorted Files, Hashing Techniques, Index Structure for files, Different types of Indexes- B-Tree - B+Tree, Query Processing.

Transaction Management: Transaction Processing, Need for Concurrency control, Desirable properties of Transaction, Schedule and Recoverability, Serializability and Schedules; Concurrency Control, Types of Locks, Two Phases locking, Deadlock, Time stamp based concurrency control , Recovery Techniques, Concepts- Immediate Update- Deferred Update, Shadow Paging.

Current Trends: Introduction to Distributed and parallel databases, Deductive Databases, Multimedia Databases, Real-Time Databases.

TEXT BOOKS, AND/OR REFERENCE MATERIAL

1. B. Desai, “An introduction to database concepts”, 3rd Edition, Galgotia publications.
2. Elmsari and Navathe, “Fundamentals of database systems”, 6th Edition, Addison Wesley.
3. J.D.Ullman, “Principals of database systems”, 8th Edition, Galgotia publications.

4. Abraham Silberschatz, Henry F. Korth and S. Sudarshan, "Database System Concepts", 4th Edition, McGraw-Hill
5. RamezElmasri and Shamkant B. Navathe, "Fundamental Database Systems", 5th Edition, Pearson Education,

DEPARTMENT: COMPUTER SCIENCE AND ENGINEERING
COURSE CODE: CSMI-202
COURSE TITLE: COMPUTER ORGANIZATION & ARCHITECTURE
COURSE DESIGNATION: REQUIRED
PRE-REQUISITES: NONE
CONTACT HOURS/CREDIT SCHEME: (L-T-P-C: 3-0-0-3)
COURSE ASSESSMENT METHODS: Two sessional exams and one end-semester exam, along with assignments, presentations and class tests which may be conducted by the course coordinator in lieu of internal assessment.

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

1. Learn the fundamentals of computer organization and its relevance to classical and modern problems of computer design.
2. Identify where, when and how enhancements of computer performance can be accomplished.
3. Learn the sufficient background necessary to read more advance texts as well as journal articles on the field.
4. See how to use concepts of computer organization in real-life settings using various PC performance improvements.

Course Outcomes	Program outcomes											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CSMI-202												
CO 1.	M	L	M	H			M					
CO 2.	M	L	L	M	M	M	H					
CO 3.	H			M		L					H	
CO 4.	L	M				H					M	

TOPICS COVERED

Introduction: Historical overview, economic trends, underlying technologies, Data Representation- Data Types, Complements. Fixed-Point Representation, Floating-Point Representation. Error Detection and Correction, Addition, Subtraction, Multiplication and Division algorithms and hardware.

Register Transfer and Micro operations: Register transfer language, Inter-Register Transfer, Arithmetic Micro-operations, Logic and Shift micro-operations Language, Control functions.

Arithmetic Logic Unit: Arithmetic, logic and shift micro operations. Constructing an arithmetic logic shift unit.

Basic Computer Architecture and Design: Computer registers, Computer Instructions-Instruction Set Completeness. Classifying Instruction Set Architecture. Basic steps of Instruction Execution, Hardwired Control, Micro programmed Control. Horizontal and Vertical Microprogramming. Interrupts.

Central Processing Unit: General Register Organization. Stack Organized CPU. Instruction Formats, Addressing Modes. Data Transfer and Manipulation. RISCVs CISC.

Memory Organization: Memory Systems: principle of locality, principles of memory hierarchy Caches, associative memory, main memory, Virtual memory, Paging and Segmentation, Memory Interleaving.

Input Output Organization: I/O performance measures, types and characteristics of I/O devices, I/O Modes-Programmed I/O, Interrupt Initiated I/O and DMA. Buses: connecting I/O devices to processor and memory, interfacing I/O devices to memory, processor, and operating system.

TEXT BOOKS, AND/OR REFERENCE MATERIAL

1. M Moris Mano, "Computer System Architecture", Pearson Education, 3rd Edition 1993.
2. David A. Patterson and John L. Hennessy, "Computer Organization & Design-The Hardware/Software Interface", Morgan Kaufmann, 2nd Edition 1997.
3. William Stallings, "Computer Organisation and Architecture, Designing for Performance", Pearson Education Asia, 6th Edition 2003.
4. Harry F. Jordan and Gita Alaghband, "Fundamentals of Parallel Processing", Pearson Education, 1st Edition 2003.
5. Barry Wilkinson Michael Allen, "Parallel Programming", prentice hall, 1999.

DEPARTMENT: COMPUTER SCIENCE AND ENGINEERING
COURSE CODE: CSMI-301
COURSE TITLE: OPERATING SYSTEM
COURSE DESIGNATION: REQUIRED
PRE-REQUISITES: NONE
CONTACT HOURS/CREDIT SCHEME :(L-T-P-C: 3-0-0-3)
COURSE ASSESSMENT METHOD: Two sessional exams and one end-semester exam, along with assignments, presentations and class tests which may be conducted by the course coordinator in lieu of internal assessment.

COURSE OUTCOMES
After the course completion, the students would be able to:

1. Analyze the working of an operating system and its components.
2. Define and analyze the synchronization process.
3. Identify the working methodology of multithreaded applications.
4. Compare and analyze different file systems being used in different operating systems.

Course Outcomes	Program outcomes											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CSMI-301												
CO 1	H	H	M	M								
CO 2		M	L	L								
CO 3		M		H	M				M			
CO 4		M		M	H							

TOPICS COVERED

Introduction to operating systems: Evolution of operating systems, Types of operating systems. The process concept, system programmer's view of processes, operating system's views of processes, operating system services for process management.

CPU Scheduling: Scheduling concepts, scheduling algorithms, algorithm evaluation, multiple processor scheduling, real time scheduling.

Deadlocks and Concurrent programming: Critical regions, Conditional critical regions, Monitors, Interprocess communication, Messages, Pipes, Semaphores, Modularization, Synchronization, Concurrent languages. Deadlocks: Characterization, Prevention, Avoidance, Detection and Recovery, Combined approach to Deadlock Handling, precedence graphs.

Memory management: Memory Management, Contiguous allocation, static-swapping, overlays, dynamic partitioned memory allocation, demand paging, page replacement, segmentation. Non-contiguous allocation, paging, Hardware support, Virtual Memory.

File systems: A Simple file system, General model of a file system, Symbolic file system, Access control verification, Logical file system, Physical file system, Allocation strategy module, Device strategy module, I/O initiators, Device handlers, Disk scheduling.

Secondary-Storage Structure: Overview of Mass-Storage Structure, Disk Structure, Disk Attachment, Disk Scheduling, Disk Management, Swap-Space Management RAID.

TEXT BOOKS, AND/OR REFERENCE MATERIAL

1. J.L. Peterson and A. Silberchatz, "Operating System Concepts", 9th Edition, Addison Wesley.
2. Harvey M. Dietel, "An Introduction to Operating System", 4th Edition, Addison Wesley.
3. W. Stallings, "Operating systems", 3rd Edition, Prentice Hall.
4. A.S. Tanenbaum, "Modern Operating system", 5th Edition, PHI
5. Dhananjay M. Dhamdhere., "Operating Systems: A Concept-Based Approach", 3rd Edition, Mc Graw Hill India.

DEPARTMENT: COMPUTER SCIENCE AND ENGINEERING
COURSE CODE: CSMI-302
COURSE TITLE: OBJECT ORIENTED PROGRAMMING
COURSE DESIGNATION: REQUIRED
PRE-REQUISITES: NONE
CONTACT HOURS/CREDIT SCHEME: (L-T-P-C: 3-0-0-3)
COURSE ASSESSMENT METHODS: Two sessional exams and one end-semester exam, along with assignments, presentations and class tests which may be conducted by the course coordinator in lieu of internal assessment.

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

1. Understand fundamentals of programming such as variables, conditional and iterative execution, methods, etc.
2. Understand fundamentals of object-oriented programming, including defining classes, invoking methods, using class libraries, etc.
3. Have the ability to write a computer program to solve specified problems.
4. Be able to use the OOP environment to create, debug and run simple C++ programs.

Course Outcomes	Program outcomes											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CSMI-302												
CO 1.	M	M	H	L	L	L						
CO 2.	M	M	H	L	L							
CO 3.	L	M	H	M					M			
CO 4.		L	H	L				M	M			

TOPICS COVERED

Object oriented thinking: Need for OOP Paradigm, Procedural programming vs object oriented programming, object oriented concepts.

Functions: Main function, function prototyping, inline functions, reference variables, call by reference, Defaults arguments, function overloading, Math library functions.
Class: Difference between C structure and class, specifying a class, Defining member functions: inside and outside class, scope resolution operator, Array within a class, array of objects, Static data members and member functions, Object as function arguments, returning objects, Friend function, memory allocation for objects ,pointer to members, pointer to object, this pointer local classes.

Constructor and destructor: Constructor, types of constructors: default, parameterized and copy constructor, constructor overloading, constructor with default parameter, dynamic initialization of objects, destructor
Operator overloading and Type Conversion: Defining operator overloading, overloading unary and binary operator, Data Conversion: Basic to User Defined , User defined to basic, Conversion from one user-defined to other.

Inheritance and polymorphism: Base class, derived class, visibility modes, derivation and friendship, Types of inheritance, Containership, virtual function binding, pure virtual functions, Abstract class, pointer to derived class.

Console IO operations: C++ stream classes, Unformatted IO operations, formatted IO operations, managing output with manipulators.
Working with files: Classes for file stream operations, opening and closing files, File opening modes, file Pointers, Error handling during file operations, command line arguments. Templates: Class template, class template with parameter, function template, function template with parameter.

TEXT BOOKS, AND/OR REFERENCE MATERIAL

1. BjraneStroustrup, "C++ Programming language",3rd edition, Pearson education Asia(1997)
2. LaforeR."Object oriented Programming in C++", 4th Ed. Techmedia,New Delhi(2002).
3. YashwantKenetkar,"Let us C++",1stEd.,Oxford University Press(2006)
4. B.A. Forouzan and R.F. Gilberg,CompilerScience,"A structured approach using C++" Cengage Learning, New Delhi.

DEPARTMENT: COMPUTER SCIENCE AND ENGINEERING
COURSE CODE: CSMI-401
COURSE TITLE: SOFTWARE ENGINEERING
COURSE DESIGNATION: REQUIRED
PRE-REQUISITES: NONE
CONTACT HOURS/CREDIT SCHEME: (L-T-P-C: 3-0-0-3)
COURSE ASSESSMENT METHODS: Two sessional exams and one end-semester exam, along with assignments, presentations and class tests which may be conducted by the course coordinator in lieu of internal assessment.

COURSE OUTCOMES
After the completion of the course, students will be able to

1. Select and implement different software development process models.
2. Extract and analyze software requirements specifications for different projects.
3. Develop some basic level of software architecture/design and apply standard coding practices.
4. Define the basic concepts and importance of Software project management concepts like cost estimation, scheduling and reviewing the progress.
5. Apply different testing and debugging techniques and analyze their effectiveness.
6. Define and analyze software risks, risk management strategies, software quality and reliability on the basis of International quality standards.

Course Outcomes	Program outcomes											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CSMI-401												
CO 1.	M	M				L						
CO 2.		H			L					H		H
CO 3.	H		H		M							
CO 4.			L	H	M				M			H
CO 5.		M	M		L							
CO 6.			M	M	H	H	H	H	H	M	M	H

TOPICS COVERED

Introduction: Problem domain, software engineering challenges and approaches, Software myths.

Software Processes: Software process, features of software process, software development process models.

Software Requirements analysis and specification: Software requirements, problem analysis, requirements specification, functional specification with use cases, validation, matrices.

Software Architecture: Role of software architect, architecture views, component and connector view, architecture style for C & C view, discussion and evaluating architectures.

Planning a software project: Effort estimation, project monitoring plan, scheduling and staffing, software configuration management plan, quality assurance plan, risk management.

Function oriented design: Design principles, module level concepts, design notation and specification, structured design methodology, verification, metrics.

Object oriented design: OO concepts, design concept, Unified Modeling Language, design methodology, metrics.

Detailed Design, Software Measurements, metrics and Models: Detailed design and PDL, verification, Metrics and their scope, Qualities of a good Software metrics, classification of metrics, Cost estimation models COCOMO, Quality attributes, SQA, Quality Standards, ISO 9000 and CMM.

Coding: Programming principles and guidelines, coding process, refactoring, verification, metrics.

Testing: A strategic methodology to software Testing fundamentals, black-box testing, white-box testing, testing process, defect analysis and prevention, metrics - reliability estimation.

TEXT BOOKS, AND/OR REFERENCE MATERIAL

1. Pankaj Jalote, "An integrated approach to software engineering", 3rd Edition, Narosa Publishing.
2. Rajib Mall, "Fundamentals of Software Engineering", 4th Edition, Pretence Hall of India.
3. Pressman Roger R, "Software Engineering: A Practitioner's Approach", 5th Edition, TMH.
4. Ian Sommerville, "Software Engineering", 9th Edition, Addison-Wesley.

DEPARTMENT: COMPUTER SCIENCE AND ENGINEERING
COURSE CODE: CSMI-402
COURSE TITLE: DATA ANALYTICS
PRE-REQUISITES: NONE
CONTACT HOURS/CREDIT SCHEME: (L-T-P-C: 3-0-0-3)
COURSE ASSESSMENT METHODS: Two sessional exams and one end-semester exam, along with assignments, presentations and class tests which may be conducted by the course coordinator in lieu of internal assessment.

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

1. Understand what Big Data is and why classical data analysis techniques are no longer adequate.
2. Understand the benefits that Big Data can offer to businesses and organizations.
3. Learn conceptually how Big Data is stored.
4. See how Big Data can be analyzed to extract knowledge.

Course Outcomes	Program outcomes											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CSMI-402												
CO 1		M	H		M				H		H	H
CO 2		H	H	M	M				H	H	M	
CO 3					H	M			H		M	
CO 4		H		H		M		H			H	H

TOPICS COVERED

Mathematical concepts in data analytics: Descriptive Statistics, Probability Distributions, Inferential Statistics, Inferential Statistics through hypothesis tests, Regression & ANOVA, Regression, ANOVA (Analysis of Variance)

Differentiating algorithmic and model based frameworks Regression: Ordinary Least Squares, Ridge Regression, Lasso Regression, K Nearest Neighbors Regression & Classification

Supervised Learning with Regression and Classification techniques -1, Bias-Variance Dichotomy, Model Validation Approaches, Logistic Regression, Linear Discriminant Analysis, Quadratic Discriminant Analysis, Regression and Classification Trees, Support Vector Machines

Supervised Learning with Regression and Classification techniques -2, Ensemble Methods: Random Forest Neural Networks, Deep learning

Unsupervised Learning and Challenges for Big Data Analytics, Clustering, Associative Rule Mining, Challenges for big data analytics

Prescriptive analytics, creating data for analytics through designed experiments, creating data for analytics through Active learning, creating data for analytics through Reinforcement learning

Hadoop Overview: Introduction to Hadoop, RDBMS vs Hadoop, key aspects of hadoop, hadoop components, hadoop conceptual layer, high level architecture of hadoop.

Hadoop Architecture: Hadoop architecture, Hadoop ecosystem components, Hadoop storage: HDFS, Hadoop processing, Map Reduce Framework, Hadoop server roles.

Hadoop big data technology landscape: NoSQL, Types of NoSQL database, Advantages, New SQL, Comparison of SQL, NoSQL and NewSQL.

TEXT BOOKS, AND/OR REFERENCE MATERIAL

1. Hastie, Trevor, et al. The elements of statistical learning. Vol. 2. No. 1. New York: Springer, 2009.
2. Montgomery, Douglas C., and George C. Runger. Applied statistics and probability for engineers. John Wiley & Sons, 2010
3. Big data and Analytics by Seema Acharya and Subhashini Chellappan.
4. Hadoop: The Definitive Guide by Tom White.
5. Big Data Analytics: From Strategic Planning to Enterprise Integration with Tools, Techniques, NoSQL andGraph by David Loshin.