Vidya Pratishthan's Kamalnayan Bajaj Institute of Engineering and Technology, Baramati

Department of Information Technology
S.Y. B. Tech Syllabus 2024-25 (As per NEP 2020)

About IT Department

- Involvement of Experts from IITs, Govt. Colleges, Reputed Industries, Alumni and Students in development of curriculum.
- Automatic Bank Credit System (ABC)
- Choice of Electives
- Remedial Teaching
- Sponsorship for Publications and IPR
- Research Mentorship
- Industry Internship
- Provision of Credit Transfer Scheme (CTS)
- Peer Teaching Scheme
- Teacher Guardian Scheme (TGS)
- Proficiency Courses
- MOUs with Industries

INSTITUTE VISION AND MISSION

VISION

To achieve Academic Excellence through Persistent and Synergic Collaborations amongst all Stakeholders.

MISSION

1. To ensure holistic development of students as lifelong learners and problem solvers through value-based quality education.

2. To motivate faculty to attain the state-of-the-art knowledge and wisdom in their domain and be a facilitator towards co creation of knowledge.

3. To frame and deploy conducive and empowering policies for multifaceted growth of students, faculty and staff to make them contributors towards excellence.

4. To partner with industry for mutually beneficial relations to generate employable and deployable workforce.

5. To fulfill the aspirations of alumni, parents, society, region and nation at large by generating technically competent and contributing manpower.

DEPARTMENT VISION AND MISSION

VISION

To develop globally competent IT professionals through continuous learning

MISSION

1. To provide graduates with the programming skills and domain knowledge.

2. To collaborate with local, state, national, and international entities in education.

3. To develop technically competent professionals with social values and ethics.

4. To encourage faculty to acquire state-of-the art knowledge.

Programme Educational Objectives (PEOs)

A graduate in IT will be able to demonstrate:

PEO1: Graduates of the program will possess strong fundamental concepts in mathematics, science, engineering and Technology to address technological challenges.

PEO2: Possess knowledge and skills in the field of Computer Science & Engineering and Information Technology for analyzing, designing and implementing complex engineering problems of any domain with innovative approaches.

PEO3: Possess an attitude and aptitude for research, entrepreneurship and higher studies in the field of Computer Science & Engineering and Information Technology.

PEO4: Have commitment to ethical practices, societal contributions through communities and lifelong learning.

PEO5: Possess better communication, presentation, time management and team work skills leading to responsible & competent professionals and will be able to address challenges in the field of IT at global level.

Program Specific Outcomes (PSOs)

At the end of the programme students will be able to demonstrate:

PSO1: Students will be able to apply engineering principles and practices for the development and maintenance of software system.

PSO2: Students will be able to undertake a team project by following professional ethical practices.

PSO3: Students will be able to make successful career in IT industry meeting the requirement of industries.

Program Outcomes (POs)

Engineering Graduates will be able to:

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.

6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Second Year (SY B. Tech.) Information Technology
w. e. f. AY:2024-2025

SEMESTER-I

Course Code	Courses Name	Teaching Scheme			Examination Scheme and Marks							Credits			
		TH	PR	TUT	Activity	ISE	ESE	TW	PR	OR	Total	TH	PR	TUT	Total
IT23201	Discrete Mathematics	3			20	20	70				110	3			3
IT23202	Fundamentals of Data Structures	3	2		20	20	70	20	20		150	3	1		4
IT23203	Logic Design and Computer Organization	3	2		20	20	70	20	20		150	3	1		4
IT23204	Database Management System	3	2		20	20	70	20	20		150	3	1		4
OE230XX	Open Elective	2					50				50	2			2
IT23205	Programming in JAVA		4					40	20		60		2		2
MD23XXX	Multi-disciplinary minor	2	2		20	20	50	20			110	2	1		3
	Total	**16**	**12**	**0**	**100**	**100**	**380**	**120**	**80**	**0**	**780**	**16**	**6**	**0**	**22**

SEMESTER-II

Course Code	Courses Name	Teaching Scheme			Examination Scheme and Marks							Credits			
		TH	PR	TUT	Activity	ISE	ESE	TW	PR	OR	Total	TH	PR	TUT	Total
GS23201	Mathematics - III	3		1		20	70	20			110	3		1	4
IT23211	Data Structures	3	2		20	20	70	20	20		150	3	1		4
IT23212	Operating System	3	2		20	20	70	20	20		150	3	1		4
IT23213	Microprocessor and Microcontroller	3	2		20	20	70	20	20		150	3	1		4
IT23214	Software Engineering	3			20	20	70				110	3			3
MD23XXX	Multi-disciplinary minor	2	2		20	20	50	20			110	2	1		3
	Total	**17**	**8**	**1**	**100**	**120**	**400**	**100**	**60**	**0**	**780**	**17**	**4**	**1**	**22**

Dr. T.V.Bhandare
HoD – IT

Dr. S.M. Bhosle
Dean Academics

Dr. R.S. Bichkar
Principal

Bucket of Multidisciplinary Minor Course

Multidisciplinary Minor Subjects	
Subject Code	**Subject Name**
AI23051	AI & Machine Learning
AI23052	Data Science
AI23053	Generative AI (Sem V+)
CO23051	Cloud Computing
CO23052	High Performance Computing (Sem V+)
CO23053	Computer Graphics & Gaming
IT23051	Cyber security
IT23052	Full Stack Development
ET23051	Embedded Systems
ET23052	Drone Technology
ET23053	Internet of Things
CE23051	Waste Management
CE23052	Green building & smart cities
ME23051	3-D Printing
ME23052	Robotics & Automation
EL23051	Solar Technology
EL23052	Industrial Automation
GS23051	Nanotechnology
GS23052	Linear Algebra and Statistics

Bucket of Open Electives

Open Elective Subjects	
Subject Code	**Subject Name**
OE23001	Digital Marketing
OE23002	Professional Leadership
OE23003	Organizational Behaviour
OE23004	Industrial Management
OE23005	Disaster Management
OE23006	Energy Economics & Management
OE23007	Operations Research
OE23008	Intellectual Property Rights
OE23009	Cyber Laws
OE23010	Bioinformatics
OE23011	Biotechnology
OE23012	International Relations
OE23013	Universal Human Values
OE23014	Education Technology
OE23015	Design Thinking
OE23016	Accounting & Finance
OE23017	Sustainability & Climate Change
OE23018	Agriculture Technology
OE23019	Architectural Technology

HONORS DEGREE
(only for students having CGPA >= 7.5)
Honor: Computational Intelligence
Honor: Cloud Computing and Virtualization, Data Science
Honor: Data Science
Honor: Artificial Intelligence
Honor: Cyber security
Honor: VLSI Design Technology
Honor: Advanced Communication Systems
Honor: Advances in Construction Technology
Honor: Advanced Structural Engg.
Honor: Robotics and Automation
Honor: Refrigeration & Air-conditioning
Honor: Renewable Energy and E- mobility

DOUBLE MINOR DEGREE
(only for students having CGPA >= 7.5)
Double Minor: Artificial Intelligence and Data Science
Double Minor: Cloud Computing and Virtualization
Double Minor: Full Stack Development
Double Minor: Embedded Systems and Real-Time OS
Double Minor: Municipal or Urban Engineering
Double Minor: Enterprise Resource Planning
Double Minor: Digital Mfg. and Robotics
Double Minor: Renewable Energy

SEMESTER I

Second Year (SY B. Tech.) INFORMATION TECHNOLOGY
w. e. f. AY:2024-2025

SEMESTER-I

Course Code	Courses Name	Teaching Scheme			Examination Scheme and Marks							Credits			
		TH	PR	TUT	Activity	ISE	ESE	TW	PR	OR	Total	TH	PR	TUT	Total
IT23201	Discrete Mathematics	3			20	20	70				110	3			3
IT23202	Fundamentals of Data Structures	3	2		20	20	70	20	20		150	3	1		4
IT23203	Logic Design and Computer Organization	3	2		20	20	70	20	20		150	3	1		4
IT23204	Database Management System	3	2		20	20	70	20	20		150	3	1		4
OE230XX	Open Elective	2					50				50	2			2
IT23205	Programming in JAVA		4					40	20		60		2		2
MD23XXX	Multi-disciplinary minor	2	2		20	20	50	20			110	2	1		3
	Total	**16**	**12**	**0**	**100**	**100**	**380**	**120**	**80**	**0**	**780**	**16**	**6**	**0**	**22**

Dr. T.V.Bhandare
HoD – IT

Dr. S.M. Bhosle
Dean Academics

Dr. R.S. Bichkar
Principal

IT23201: DISCRETE MATHEMATICS

Course Code : IT23201	Course Credits: 03	Course Type: PCC

Teaching Scheme			Evaluation Scheme					
TH	PR	TUT	ACTIVITY	ISE	ESE	TW	PR	OR
3	--	-	20	20	70	--	-	--

Prerequisite Course Mapping:
1. Basic Mathematics

Future Course Mapping:
1. Design and Analysis of Algorithms

Importance of Course: Concepts and notations from discrete mathematics are useful in studying and describing objects and problems in branches of computer science, such as computer algorithms, programming languages, cryptography, automated theorem proving, and software development.

Course Objectives:
1. To gain sound knowledge to formulate and solve problems with sets and propositions.
1) To understand and solve counting problems by applying elementary counting techniques to solve problems of discrete probability.
2. To understand Graph and Tree terminologies and models to be applied in real life problems.
3. To recognize types of relation, formulate and solve problems with relations and functions.
4. To understand the basics of number theory and its applications.
2) 6. To understand the various types' algebraic structures and its applications

Course Outcomes:
1) **Formulate** and apply formal proof techniques and solve the problems with logical reasoning.
2) **Analyze and evaluate** the combinatorial problems by using probability theory.
3) **Apply** the concepts of graph theory to devise mathematical models.
4) **Analyze** types of relations and functions to provide solutions to computational problems.
5) **Identify** techniques of number theory and its application.
6) **Identify** fundamental algebraic structures.

UNIT No.	Syllabus	Teaching Hours

I	**Sets**: Sets, Combinations of Sets, Venn Diagram, Finite and Infinite Sets, Countable Sets, Multisets, Cartesian Product, Principle of Inclusion and Exclusion, Mathematical Induction. **Propositions**: Propositions, Logical Connectives, Conditional and Bi-conditional. Propositions, Logical Equivalence, Validity of Arguments by using Truth Tables, Predicates and Quantifiers, Applications of propositional logic	6
II	**Combinatorics:** Rules of Sum and Product, Permutations, Combinations. **Discrete Probability**: Discrete Probability, Conditional Probability, Bayes Theorem, Applications of Combinatorics and Discrete Probability.	6
III	**Relations:** Properties of Binary Relations, Closure of Relations, Warshall's Algorithm, Equivalence Relations, Partitions, Partial Ordering Relations, Lattices. **Functions:** Functions, Composition of Functions, Invertible Functions, Pigeonhole Principle.	6
IV	**Graphs:** Basic Terminologies, Multi-Graphs, Weighted Graphs, Sub Graphs, Isomorphic graphs, Complete Graphs, Regular Graphs, Bipartite Graphs, Operations on Graphs, Paths, Circuits, Hamiltonian and Eulerian graphs, Travelling Salesman Problem, Dijkstra's Algorithm, Planar Graphs, Graph Coloring.	6
V	**Trees:** Tree Terminologies, Rooted Trees, Properties, Prefix Codes, Spanning Trees, Fundamental Cut Sets and Circuits, Max flow –Min Cut Theorem (Transport Network). Binary Trees and traversal, Binary Search Tree and Traversals.	6
VI	**Algebraic Structures:** Introduction Semigroup, Monoid, Group, Abelian Group, Permutation Groups, Ring, Integral Domain, Field. Applications of Algebraic Structures.	6

Text Books :
1. Kenneth H. Rosen. Discrete Mathematics and Its Applications, 7th Edition, McGraw Hill, 2012.
2. C. L. Liu and D. P. Mohapatra, "Elements of Discrete Mathematics", 4th Edition, McGraw-Hill.

Reference Books :
1. Bernard Kolman, Robert C. Busby, Sharon Cutler Ross, "Discrete mathematical structures", 6th edition, Prentice Hall of India.
2. Edgar G. Goodaire, Michael M. Parmenter, "Discrete Mathematics with Graph Theory", 3rd Edition, Pearson Education.
3. Tremblay J. S., "Discrete mathematical structures with application", 3rdEdition, Tata McGraw Hill.
4. Lipschutz Seymour, "Discrete mathematics", 4th Edition, Tata McGraw-Hill.
5. Johnsonbaugh Richard, "Discrete Mathematics", 7th edition, Pearson.
6. Narsingh Deo, "Graph Theory with Applications to Engineering and Computer Science", Dover publication

Online Resources:
1. NPTEL Course "Discrete Mathematics" By Prof. Sudarshan Iyengar, Prof. Neeldhara | IIT Ropar, IIT Gandhinagar.
2. https://onlinecourses.nptel.ac.in/noc20_cs37/unit?unit=41&lesson=42

IT23202: FUNDAMENTALS OF DATA STRUCTURES

Course Code : IT23202	Course Credits: 04	Course type: PCC

Teaching Scheme			Evaluation Scheme					
TH	PR	TUT	ACTIVITY	ISE	ESE	TW	PR	OR
3	2		20	20	70	20	20	0

Prerequisite Course Mapping:
1. Object Oriented Programming

Future Course Mapping:
1. Data Structures

Importance of Course: This course will be useful in getting technical knowledge that will be required for software building and in day to day life. Data structure used in Google map and in searching or sorting data.

Course Objectives:
1. To learn algorithm development and analysis of algorithms.
2. To learn different searching and sorting techniques.
3. To implement dynamic memory allocation technique
4. To build the logic to use appropriate data structure in logical and computational solutions.
5. To implement stack and queue data structure and understand its applications.
6. To build the logic to use appropriate data structure in logical and computational solutions,

Course Outcomes:
1. **Analyze** algorithms with respect to time and space complexity
2. **Apply** appropriate searching and/or sorting techniques in the application development.
3. **Implement** dynamic memory allocation technique to solve computational problems.
4. **Demonstrate** use of sequential data structures-Linked lists to store and process data.
5. **Implement** stack data structure and **Understand** its applications.
6. **Understand** and **Implement** various types of queue

UNIT No.	Syllabus	Teaching Hours
I	**Introduction** **Introduction to Data Structures**: Concept of data, Data object, Data structure, Concept of Primitive and non-primitive, linear and Nonlinear, static and dynamic, persistent and ephemeral data structures, Definition of ADT **Algorithms**: Problem Solving, Introduction to algorithm, Characteristics of algorithm, Algorithm design tools: Pseudo code and flowchart. Complexity of algorithm: Space complexity, Time complexity,	6

		Asymptotic notation- Big-O, Theta and Omega, finding complexity using step count method	
II		**Searching and Sorting** **Searching**: Search Techniques-Sequential Search/Linear Search, Variant of Sequential Search- Binary Search, Variant of Sequential Search- Indexed sequential search, Sentinel Search **Sorting**: Types of Sorting-Internal and External Sorting, General Sort Concepts-Sort Order, Stability and Efficiency, Comparison Based Sorting Methods-Bubble Sort, Quick Sort, Shell Sort, Merge Sort. Comparison Based Sorting Methods -Insertion Sort, Selection Sort Non-comparison Based Sorting Methods- Radix sort, Counting sort, Bucket sort	6
III		**Pointers and File Handling in C/C++** Introduction to Pointers, pointer to pointer, pointer to single and multidimensional arrays and address calculation, sparse matrix, array of pointers, pointer to functions, dynamic memory allocation.	6
IV		**Linear Data Structure using Linked Representation** Concept of linked organization, Singly Linked List, Doubly Linked List, Circular Linked List (Operations: Create, Display, Search, Insert, Delete).Realization of linked list using dynamic memory management, Doubly circular linked list, sort and concatenation of linked list, Polynomial Manipulations-Polynomial addition. Generalized Linked List (GLL) concept, Representation of Polynomial using GLL	6
V		**Stack** Concept of stack, stack as an ADT using sequential and linked organization, Applications of stack: Recursion- concept, variants of recursion, converting expressions from infix to postfix or prefix form, evaluating postfix or prefix form, backtracking algorithmic strategy, use of stack in backtracking.	6
VI		**Queue** Concept of queues as ADT, Implementation of queue using array and linked organization, Concept of circular queue, double ended queue, Priority Queue- Basic concept, types (Ascending and Descending).	6

List of Practical Assignments:
1. Represent sets using one-dimensional arrays and implement functions to perform:
   a. Union
   b. Intersection
   c. Difference
   d. Symmetric difference of two sets
2. Represent matrix using two-dimensional arrays and perform the following operations with and without using pointers:
   a. Addition
   b. Multiplication
   c. Transpose

3. Implement a sparse matrix with operations like initialize an empty sparse matrix, insert an element, add two matrices and transpose a matrix.
4. Implement a Polynomial with operations like create an empty polynomial, insert an entry into a polynomial, add two polynomials and evaluate a polynomial.
5. Consider a student database of SEIT class (at least 15 records). Database contains different fields of every student like Roll No, Name and SGPA.(array of structure).
   a. Arrange a list of students alphabetically. (Use Bubble sort/Insertion sort)
   b. Arrange a list of students to find out first ten toppers from a class. (Use Quick sort/Radix sort)
   c. Search a particular student according to name using binary search without recursion. (all the student records having the presence of search key should be displayed)
6. The Department of Information Technology has a student's club named 'Research Club'. Students of second, third and final year of department can be granted membership on request. Similarly one may cancel the membership of a club. First node is reserved for the president of the club and the last node is reserved for the secretary of the club. Write program to maintain club member's information using singly linked lists. Store student PRN and Name. Write functions to:
   a. Add and delete the members as well as president or even secretary.
   b. Compute total number of members of club
   c. Display members
   d. Two linked lists exist for two divisions. Concatenate two lists.
7. Implement stack as an abstract data type using singly linked list
8. Use stack as ADT for conversion of infix expression to postfix, prefix.
9. Implement Queue using Linked List. Perform following operations on it.
   a. Insertion (Enqueue)
   b. Deletion (Dequeue)
   c. Display
10. Implement Circular Queue using Array. Perform following operations on it.
   a. Insertion (Enqueue)
   b. Deletion (Dequeue)
   c. Display (Note: Handle queue full condition by considering a fixed size of a queue.)

Text Books :
1. E. Horowitz, S. Sahni, D. Mehta, "Fundamentals of Data Structures in C++", Galgotia Book Source, New Delhi, 1995, ISBN 16782928
2. Y. Langsam, M. Augenstin, A. Tannenbaum, "Data Structures using C and C++", 2nd Edition, Prentice Hall of India, 2002, ISBN-81-203-1177-9

Reference Books:
1. G. A.V, PAI , "Data Structures and Algorithms ", McGraw Hill, ISBN -13: 978-0-07-066726-6
2. M. Folk, B. Zoellick, G. Riccardi, "File Structure And Object Oriented Approach with C++", Pearson Education, 2002, ISBN 81 - 7808 - 131 - 8.
3. Sartaj Sahni. "Data Structures, Algorithms and Applications" in C++. Second edition, university press ISBN-13: 978-0072362268.

Online Resources:
1. https://nptel.ac.in/courses/106/102/106102064/
2. https://nptel.ac.in/courses/106/105/106105085
3. https:// nptel.ac.in/courses/106/106/106106127
4. https://www.ebookphp.com/advanced-data-structures-epub-pdf/

5. https://www.ebookphp.com/data-structures-and-algorithms-professional-edition-beginners-guide-epubpdf/

IT23203 : LOGIC DESIGN AND COMPUTER ORGANIZATION

Course Code: IT23203		Course Credits: 04			Course type: PCC			
Teaching Scheme			**Evaluation Scheme**					
TH	PR	TUT	ACTIVITY	ISE	ESE	TW	PR	OR
3	02	–	20	20	70	20	20	-

Prerequisite Course Mapping:
1. Basics of electronics engineering

Future Course Mapping:
1. Microprocessor and Microcontroller

Importance of Course: Digital circuit design (Combinational and Sequential circuit) has various applications in the field of Electronics and Digital device manufacturing.

Course Objectives:
1. To make undergraduates aware of different levels of abstraction of computer systems from a hardware perspective.
2. To make undergraduates understand the functions, characteristics of various components of Computer, in particular processor & memory.

Course Outcomes:
1. Perform basic binary arithmetic & simplify logic expressions.
2. Grasp the operations of logic ICs and implement combinational logic functions using ICs
3. Comprehend the operations of basic memory cell types and implement sequential logic functions using ICs.
4. Elucidate the functions & organization of various blocks of CPU.
5. Understand CPU instruction characteristics, enhancement features of CPU.
6. Describe an assortment of memory types (with their characteristics) used in computer systems and basic principle of interfacing input, output devices.

Unit No.	Syllabus	Teaching Hours
I	**Introduction to Digital Electronics** Classification of logic families, Characteristics of digital ICs, Operation of TTL NAND gate, CMOS logic – CMOS NAND, Comparison of TTL & CMOS, Sign Magnitude, 1's complement & 2's complement representation, unsigned binary arithmetic, subtraction using 2's complement; **Codes:** Binary, BCD, Octal, Hexadecimal, Excess-3, Gray code, **Logic minimization:** Standard representations for logic functions, k-map representation of logic functions (SOP and POS forms), Simplification of logical functions using K-Maps up to 4 variables.	6

II	**Combinational Logic Design** **Design Combinational Logic:** Code converters, Half- Adder, Full Adder, Half Subtractor, Full Subtractor, n-bit Binary adder, Multiplexer, Demultiplexer, Decoder, Binary adder, BCD adder, BCD subtractor, combinational logic designs using MUX and Demux/Decoder.	6
III	**Sequential Logic Design** 1-bit Memory element, Latch, Flip-Flops, Excitation table of Flip-Flops, Conversion of Flip-Flops **Counters:** asynchronous, synchronous and modulus N counters **Applications of flip-flops:** Registers and counters	6
IV	**Computer Organization & Processor** Computer organization & computer architecture(typical organization ,Functions , Types), Memory, I/O & system bus, Von Neumann & Harvard architecture, Instruction cycle Processor: Single bus organization of CPU, ALU, Register: address registers, data registers, flags, PC, MAR, MBR, IR & control unit, micro Operations and control signals.	6
V	**Processor Instructions & Processor Enhancements** Instruction : elements of machine instruction, instruction representation, Instruction Format & 0-1-2-3 address formats, Types of operands Addressing modes, Instruction types based on operations, key characteristics of RISC & CISC, Interrupt: its purpose, types, classes & interrupt handling (ISR, multiple interrupts), instruction pipelining, Multiprocessor systems: Processor Architectures, types of MIMD & multicore processor, typical features of multi core:- Intel core i7.	6
VI	**Memory & Input / Output Systems** **Memory Systems:** Characteristics of memory systems, Memory hierarchy, signals to connect memory to processor, memory read & write cycle, characteristics of semiconductor memory: SRAM, DRAM & ROM, Cache memory, **Input / Output Systems:** I/O Module, Programmed I/O, Interrupt Driven I/O, Direct Memory Access (DMA).	6

List of Practical Assignments:
1. Design and implement 4-bit BCD to Excess-3 code convertor.
2. Design and implement 1 digit BCD adder using IC7483.
3. Design and implement following using multiplexer IC 74153
   1) Full adder 2) Any three variable functions (cascade method).
4. Design and implement Full subtractor using decoder IC 74138.
5. Design and implement 3 bit Up-Down Asynchronous Counter using IC 7476.
6. Design and implement 3 bit Up-Down Synchronous Counter using 7476.
7. Design and implement Modulo 'N' counter using IC7490.
8. Design & simulate ALU with four functions (AND, OR, XOR, ADD).

Text Books:
1. "Modern Digital Electronics", R.P. Jain, Tata McGraw-Hill, Third Edition.
2. "Computer organization and architecture, designing for performance" by William Stallings, Prentice Hall, Eighth edition.

Reference Books:
1. "Digital Design", M Morris Mano, Prentice Hall, Third Edition.
2. "Computer organization", Hamacher and Zaky, Fifth Edition.
3. "Computer Organization and Design: The Hardware Software Interface" D. Patterson, J. Hennessy, Fourth Edition, Morgan Kaufmann.
4. "Microprocessors and interfacing-programming and hardware" Douglas V. Hall and SSSP Rao, McGraw-Hill, Third Edition.

Online Resources:
NPTEL Course :
1.https://archive.nptel.ac.in/courses/106/105/106105185/#
2. https://archive.nptel.ac.in/courses/106/105/106105163/
Virtual Lab:
simulator Link http://vlabs.iitkgp.ac.in/coa/

IT23204: DATABASE MANAGEMENT SYSTEM

Course Code : IT23204	Course Credits: 04	Course type: PCC

Teaching Scheme			Evaluation Scheme					
TH	PR	TUT	ACTIVITY	ISE	ESE	TW	PR	OR
03	02 hrs/ week	–	20	20	70	20	20	

Prerequisite Course Mapping:
1. Discrete Mathematics
2. Data Structures and Algorithm

Future Course Mapping:
1. Advanced Database Management System
2. Data Science and Big Data Analytics

Importance of Course: Any Real Life Project Database will be created using the knowledge of DBMS.

Course Objectives:
1. To realize the fundamental concepts of Database Systems
2. To discover the various RDBMS Tools
3. To absorb different PL-SQL Types
4. To grasp how Queries will be formulated
5. To greet the concepts of Transaction Processing, Concurrency and Recovery in multi-user database environments.
6. To grow up on recent Database Trends

Course Outcomes:
1. **Understand** fundamental elements of database management systems
2. **Understand and remember** PL/SQL queries.
3. **Formulate** SQL queries on data for relational databases
4. **Understand** advanced architecture of DBMS such as Data Warehousing and Data Mining
5. **Apply** ACID properties for transaction management and concurrency control
6. **Analyze** various database architectures and technologies

Unit No.	Syllabus	Teaching Hours
1	**Introduction to DBMS** Basic concepts, Advantages of DBMS over file processing systems, Data abstraction, Database languages, Data models, Data independence, Components of a DBMS, Overall structure of DBMS, Multi-user DBMS architecture, System catalogs, ER Diagram: Basic concepts, Entity,	6

	attributes, relationships, constraints, keys, EER Diagram-Basic concepts ,characteristics	
2	**Introduction to SQL and PL/SQL** **SQL:** DDL, DML, DCL, TCL, SQL Operators. Tables: Creating, Modifying, Deleting, Updating. SQL DML Queries: SELECT Query and clauses, Index and Sequence in SQL. Views: Creating, Dropping, Updating using Indexes, Set Operations, Predicates and Joins, Set membership, Tuple Variables, Set comparison, Ordering of Tuples, Aggregate Functions, SQL Functions, Nested Queries **PL/SQL:** Concept of Stored Procedures and Functions, Cursors, Triggers, Assertions, Roles and Privileges	6
3	**Introduction to Relational Database and Query Processing** **Relational Model:** Basic concepts, Attributes and Domains, CODD's Rules. Relational Integrity: Domain, Referential Integrities, Enterprise Constraints. Database Design: Features of Good Relational Designs, Normalization, Atomic Domains and First Normal Form, Decomposition using Functional Dependencies, Algorithms for Decomposition, 2NF, 3NF, BCNF. **Query Processing:** Introduction to Query Processing: Overview, Measures of Query cost, Selection and Join operations, Evaluation of Expressions Introduction to Query optimization: Estimation, Transformation of Relational Expression	6
4	**Introduction to Data Warehousing and Data Mining** Data Warehousing, Architecture and features of Data Warehouse, ETL Process, OLAP. Data Mining, Knowledge Discovery, Data Mining techniques: Apriori Algorithm, Decision Tree, Applications of data mining.	6
5	**Database Transaction Management** Introduction to Database Transaction, Transaction states, ACID properties, Concept of Schedule, Serial Schedule. Serializability: Conflict and View, Cascaded Aborts, Recoverable and Non-recoverable Schedules. Concurrency Control: Lock-based, Time-stamp based Deadlock handling. Recovery methods: Shadow-Paging and Log-Based Recovery, Checkpoints. Log-Based Recovery: Deferred Database Modifications and Immediate Database Modifications	6
6	**Introduction to Database Connectivity and Advances in DBMS** Web application development using Database connectivity, Database Architectures: Centralized and Client-Server Architectures, 2 Tier and 3 Tier Architecture, Introduction to Parallel Databases, Key elements of Parallel Database Processing, Architecture of Parallel Databases, Introduction to Distributed Databases, Architecture of Distributed Databases, Distributed Database Design. Emerging Database Technologies: Introduction, No SQL Databases- Internet Databases, Cloud databases, Mobile Databases, SQLite database, XML databases	6
PRACTICAL ASSIGNMENTS		

Assign No	Title
1	Study the installation and uninstallation of MySQL and SQLite
2	**ER Modeling and Normalization:** Decide a case study related to real time application in a group of 2-3 students and formulate a problem statement for application to be developed. Propose a Conceptual Design using ER features using tools like ERD plus, ER Win etc. (Identifying entities, relationships between entities, attributes, keys, cardinalities, generalization, specialization etc.) Convert the ER diagram into relational tables and normalize Relational data model. Note: Student groups are required to continue the same problem statement throughout all the assignments in order to design and develop an application as a part Mini Project. Further Assignments will be useful for students to develop a backend for the system. To design front end interface students should use the different concepts learnt in the other subjects also.
3	**SQL Queries:** a. Design and Develop SQL DDL statements which demonstrate the use of SQL objects such as Table, View, Index, Sequence, Synonym, different constraints etc. b. Write at least 10 SQL queries on the suitable database application using SQL DML Statements. Note: Instructor will design the queries which demonstrate the use of concepts like Insert, Select, Update, Delete with operators, functions, and set operator etc
4	**SQL Queries** – all types of Join, Sub-Query and View Write at least10 SQL queries for suitable database applications using SQL DML statements. Note: Instructor will design the queries which demonstrate the use of concepts like all types of Join ,Sub-Query and View
5	**Unnamed PL/SQL code block**: Use of Control structure and Exception handling is mandatory. Consider Tables: 1. Borrower(Roll_no, Name, Date of Issue, Name of Book, Status) 2. Fine(Roll_no, Date, Amt) • Accept Roll_no and Name of Book from user. • Check the number of days (from date of issue). • If days are between 15 to 30 then the fine amount will be Rs 5 per day. • If no. of days>30, per day fine will be Rs 50 per day and for days less than 30, Rs. 5 per Day
6	Write a PL/SQL code block to calculate the area of a circle for a value of radius varying from 5 to 9. Store the radius and the corresponding values of calculated area in an empty table named areas, consisting of two columns, radius and area
7	**Cursors:**(All types: Implicit, Explicit, Cursor FOR Loop, Parameterized Cursor) Write a PL/SQL code to implement all types of cursor and display empno, name and salary of 5 highest paid employees using cursor. Employee(empno,ename,join_date,designation,salary)

8	**Database Trigger** (All Types: Row level and Statement level triggers, Before and After Triggers). Write a database trigger on the Library table. The System should keep track of the records that are being updated or deleted. The old value of updated or deleted records should be added in Library_Audit table. **Note:** Instructor will Frame the problem statement for writing PL/SQL block for all types of Triggers in line with above statement.
9	**Database Connectivity:** Write a program to implement MySQL/Oracle database connectivity with any front end language to implement Database navigation operations (add, delete, edit etc.)
10	**Mini Project** Consider any database application and implement MYSQL database connectivity with Java/PHP/Python. Students can use the same problem statement decided for Assignment no-2

Text Books :
1."Database System Concepts" by Silberschatz A., Korth H., Sudarshan S. , 6th edition
2.Data Mining :Concepts and Techniques ,Jiawei Han, Micheline Kamber, Jian Pei

Reference Books:
1. C J Date, "An Introduction to Database Systems", Addison-Wesley, ISBN: 0201144719
2. S.K.Singh, "Database Systems: Concepts, Design and Application", Pearson Education, ISBN 978-81-317-6092-5
3. Joy A. Kreibich, "Using SQLite", O'REILLY, ISBN: 13:978-93-5110-934-1
4. Ivan Bayross, "SQL, PL/SQL the Programming Language of Oracle", BPB Publications ISBN: 9788176569644, 9788176569644
5. Reema Thareja, Data warehousing, Oxford University Press. ISBN 0195699610

Online Resources:
1.NPTEL Course "**Introduction to DBMS**"
https://nptel.ac.in/courses/106/105/106105175/

IT23205: Vocational and Skill Enhancement Course (VSEC) PROGRAMMING IN JAVA

Course Code: IT23205		Course Credits: 02			Course type: VSEC			
Teaching Scheme			**Evaluation Scheme**					
TH	PR	TUT	ACTIVITY	ISE	ESE	TW	PR	OR
	04 hrs/ week	–	.	.	.	40	20	-

Prerequisite Course Mapping:
1. C, C++ programming

Future Course Mapping:
1. Advanced Java Programming

Importance of Course: Java is an object oriented, secure and platform independent language so it is used in many IT industries for development of various applications such as windows, database, networking, mobile and distributed applications.

Course Objectives:
1. To apply concepts of object oriented paradigm.
2. To design and implement models for real life problems by using object-oriented programming.
3. To develop object oriented programming skills.

Course Outcomes:
1. **Understand** the basic principles of JAVA programming language.
2. **Develop** the console based java applications using classes and object.
3. **Develop** the java programs by using the concepts of methods & Inheritance.
4. **Apply** the concepts of interfaces & packages for program implementation.
5. **Apply** the multithreading and Exception handling in Java to develop robust programs.
6. **Develop** the real world application using input/output file handling.

LIST OF PRACTICAL ASSIGNMENTS

1	Design a class 'Complex 'with data members for real and imaginary part. Provide default and Parameterized constructors. Write a program to perform arithmetic operations of two complex numbers.
2	Identify commonalities and differences between Publication, Book and Magazine classes. Title, Price, Copies are common instance variables and saleCopy is common method. The differences are, Bookclass has author and orderCopies(). Magazine Class has methods orderQty, Current issue, receiveissue().Write a program to find how many copies of the given books are ordered and display total sale of publication.
3	Identify commonalities and differences between Publication, Book and Magazine classes. Title, Price, Copies are common instance variables and saleCopy is common

	method. The differences are, Bookclass has author and orderCopies(). Magazine Class has methods orderQty, Current issue, receiveissue().Write a program to find how many copies of the given books are ordered and display total sale of publication.
4	Design and develop inheritance for a given case study, identify objects and relationships and implement inheritance wherever applicable. Employee class hasEmp_name, Emp_id, Address, Mail_id, and Mobile_noas members. Inherit the classes: Programmer, Team Lead, Assistant Project Manager and Project Manager from employee class. Add Basic Pay (BP) as the member of all the inherited classes with 97% of BP as DA, 10 % of BP as HRA, 12% of BP as PF, 0.1% of BP for staff club fund. Generate pay slips for the employees with their gross and net salary.
5	Design a base class shape with two double type values and member functions to input the data and compute_area() for calculating area of shape. Derive two classes: triangle and rectangle. Make compute_area() as abstract function and redefine this function in the derived class to suit their requirements. Write a program that accepts dimensions of triangle/rectangle and display calculated area. Implement dynamic binding for given case study
6	Design and develop a context for given case study and implement an interface for Vehicles Consider the example of vehicles like bicycle, car and bike. All Vehicles have common functionalities such as Gear Change, Speed up and apply breaks. Make an interface and put all these common functionalities. Bicycle, Bike, Car classes should be implemented for all these functionalities in their own class in their own way
7	Implement a program to handle Arithmetic exception, Array Index Out of Bounds. The user enters two numbers Num1 and Num2. The division of Num1 and Num2 is displayed. If Num1 and Num2 are not integers, the program would throw a Number Format Exception. If Num2 were zero, the program would throw an Arithmetic Exception. Display the exception.
8	Implement a generic program using any collection class to count the number of elements in a collection that have a specific property such as even numbers, odd number, prime number and palindromes.
9	Implement a program for maintaining a database of student records using Files. Student has Student_id , name, Roll_no, Class, marks and address. Display the data for few students. 1. Create Database 2. Display Database 3. Delete Records 4. Update Record 5. Search Record.
10	Using concepts of Object-Oriented programming develop application for Bank system. 1. Create an account 2. Deposit money 3. Withdraw money 4. Honor daily withdrawal limit 5. Check the balance 6. Display Account information.

Text Books:
1. Herbert Schildt, "Java: The complete reference", Tata McGraw Hill, 7th Edition.
2. E. Balaguruswamy, "Programming with JAVA: A Primer" McGraw Hill Education, India, 5th Edition.

Reference Books:
1. T. Budd, "Understanding OOP with Java", Pearson Education, 2nd Updated Edition.

2. Y. Daniel Liang (2010), "Introduction to Java programming", Pearson Education, India, 7th Edition.
3. Cay Horstmann, "Core Java Volume 1", Kindle, 11th Edition.

Online Resources:
NPTEL Course :
   1. https://archive.nptel.ac.in/courses/106/105/106105185/#
   2. https://archive.nptel.ac.in/courses/106/105/106105163/
Virtual Lab:
   1. simulator Link http://vlabs.iitkgp.ac.in/coa/

SEMESTER II

Second Year (SY B. Tech.) Information Technology
w. e. f. AY:2024-2025

Course Code	Courses Name	Teaching Scheme			Examination Scheme and Marks							Credits			
		TH	PR	TUT	Activity	ISE	ESE	TW	PR	OR	Total	TH	PR	TUT	Total
GS23201	Mathematics - III	3		1		20	70	20			110	3		1	4
IT23211	Data Structures	3	2		20	20	70	20	20		150	3	1		4
IT23212	Operating System	3	2		20	20	70	20	20		150	3	1		4
IT23213	Microprocessor and Microcontroller	3	2		20	20	70	20	20		150	3	1		4
IT23214	Software Engineering	3			20	20	70				110	3			3
MD23XXX	Multi-disciplinary minor	2	2		20	20	50	20			110	2	1		3
	Total	17	8	1	100	120	400	100	60	0	780	17	4	1	22

Dr. T.V.Bhandare
HoD – IT

Dr. S.M. Bhosle
Dean Academics

Dr. R.S. Bichkar
Principal

GS23201: Mathematics - III

Teaching Scheme: Theory: 3 Hours/Week Term work: 1 Hour/Week	Credits 04	Examination Scheme: In-Semester: 20 Marks End-Semester:70 Marks Term Work: 20 Marks

Prerequisites: Differential & Integral calculus, Taylor series, Differential equations of first order and first degree, Fourier series, Collection, Classification & Representation of data.

Course Objectives:
To provide the students with concepts and techniques in Linear differential equations, Fourier transform, Statistical methods, and Probability theory. The aim is to equip them with the techniques to understand advanced-level mathematics and its applications that would be useful in their discipline and enhance their thinking power.

Course Outcomes (COs): The students will be able to learn:
CO1: Solve higher-order linear differential equations using appropriate techniques useful for modeling in their field.
CO2: Understand the concepts of Fourier transform.
CO3: Understand and apply the various concepts of statistical methods of correlation, and regression and Apply them in their field.
CO4: Apply the concepts of appropriate Probability and Probability distribution for data analysis and predictions in multiple data sets.
CO5: Solve Algebraic, Transcendental equations and System of linear equations using numerical techniques.
CO6: Compute Interpolating polynomials, numerical differentiation, and integration, numerical solutions of ordinary differential equations used in modern scientific computing.

Course Contents

Unit I: Linear Differential Equations (LDE) and Applications (07 Hours)
Introduction, Solution of LDE, General method, short-cut method, Method of variation of parameters, Cauchy's, Legendre's DE, Simultaneous DE.

Unit II: Fourier Transform and Statistics (07 Hours)
Fourier Transform: General Fourier, Fourier Sine, Cosine, and inverse transforms.
Statistics: Measures of dispersion, Moments, Skewness and Kurtosis, Correlation and Regression analysis.

Unit III: Regression Models (07 Hours)
Importance of Regression in Data Mining, Simple Linear Regression, Model: $Y = \beta 0 + \beta 1 X + \epsilon$ Assumptions, Estimation of $\beta 0$ and $\beta 1$ by the method of least squares, Multiple linear regression model $Y = \beta 0 + \beta 1 X 1 + ... + \beta p X p + \epsilon$, residuals, Least-Squares Estimation of the Regression Coefficients, obtaining normal equations, Solutions of normal equations, Generalized linear models, and applications.

Unit IV: Probability and Probability Distributions (07 Hours)
Theorems on probability, Random variables, Probability Mass function, Probability Density function, Mathematical Expectation. Binomial, Poisson, and Normal distribution and applications.

Unit V: Numerical Methods (07 Hours)
Numerical Solution of Algebraic and Transcendental Equations: Bisection, Secant, Regula-Falsi, Newton–Raphson and Successive Approximation Methods.
Numerical Solutions of System of linear equations: Gauss elimination, LU Decomposition, Cholesky, Jacobi and Gauss-Seidel Methods.

Unit VI: Numerical Methods (07 Hours)

Interpolation: Finite Differences, Newton's and Lagrange's Interpolation formula, Numerical Differentiation.

Numerical Integration: Trapezoidal and Simpson's rules.

Solution of Ordinary differential equations: Euler's, Modified Euler's, Runge-Kutta 4th order.

List of Books:

Textbooks:

1. Higher Engineering Mathematics by B.V. Ramana (Tata McGraw-Hill).
2. Higher Engineering Mathematics by B. S. Grewal (Khanna Publication, Delhi).

Reference Books:

1. Erwin Kreyszig, "Advanced Engineering Mathematics", 10ed, Wiley India
2. M. D. Greenberg, "Advanced Engineering Mathematics", 2nd e Pearson Education
3. Peter V. O'Neil, "Advanced Engineering Mathematics", 7ed, Cengage Learning
4. S. L. Ross, "Differential Equations", 3e, Wiley India
5. Sheldon M. Ross, "Introduction to Probability and Statistics for Engineers and Scientists", 5e, Elsevier Academic Press
6. M. K. Jain, S. R. K. Iyengar, and R. K. Jain, "Numerical Methods for Scientific and Engineering Computation", 5e, (New Age International Publication).
7. Draper, N. R. and Smith, H. "Applied Regression analysis", (1998) (John Wiley) Third Edition.
8. S.P. Gupta, Sultan Chand and Sons, "Statistical Methods", New Delhi, 2009.

Guidelines for Term Work Assessment

1. Tutorials for the subject shall be engaged in a minimum of three batches (batch size of 22 students) per division.
2. Term work shall consist of six assignments on each unit and is based on performance and continuous internal assessment.

IT23211: DATA STRUCTURES

Course Code : IT23211			Course Credits: 04				Course type: PCC	
Teaching Scheme			**Evaluation Scheme**					
TH	PR	TUT	ACTIVITY	ISE	ESE	TW	PR	OR
3	2	-	20	20	70	20	20	-

Prerequisite Course Mapping:
1. Fundamentals of Data Structures

Future Course Mapping:
1. Operating System
2. Database Management Systems

Importance of Course: This course will be useful in getting technical knowledge that will be required for software building and in day to day life.

Course Objectives:
1. To understand advanced data structures to solve complex problems in various domains.
2. To build the logic to use appropriate data structure in logical and computational solutions.
3. To understand various algorithmic strategies to approach the problem solution.
4. To suggest appropriate data structure and algorithm for graphical solutions of the problems.
5. To learn and understand representation, implementation and applications of trees, search trees, graphs, heaps.
6. To study different types of file organizations.

Course Outcomes:
1. **Apply** and **analyze** nonlinear data structures to solve real world complex problem
2. **Design and Apply** the non-linear data structures to
3. **Design and Implement** algorithms for graphical solutions of the problems.
4. **Identify and Apply an efficient** hashing scheme and collision handling technique for real world applications.
5. **Study and Implement** efficient multiway search techniques and Heap to store and maintain data.
6. To **Learn** different file organizations and **Apply** file handling technique for real world applications.

UNIT No.	Syllabus	Teaching Hours
I	**Trees** Introduction to Non Linear Data Structure, Binary Trees, Types of Binary Trees, Properties of Binary Trees, Binary Tree as Abstract Data Type, Representation using Sequential and Linked Organization, Binary Tree creation, Recursive and Non Recursive Tree Traversals, Threaded	6

	Binary Tree and operations, Traversals of Inorder Threaded Binary Tree, Applications of Binary Trees.	
II	**Search Trees** Binary Search Tree and its operations, Binary Search Trees as Abstract Data Type Weight balanced tree - Optimal Binary Search Tree (OBST), Height Balanced Tree- AVL tree. Red-Black Tree	6
III	**Graphs** Basic Terminologies, Storage Representation, Graph Traversals, Graph as Abstract Data Type, Spanning Trees, Minimum Spanning Trees, Kruskal's Algorithm, Prim's Algorithm, Dijkstra's Single Source Shortest Path Algorithm, Topological Sorting.	6
IV	**Hashing** General Idea of Hashing, Hash Table, Hash function: division, multiplication, extraction, mid-square, folding, Properties of good hash function, Rehashing, collision, probe, synonym, overflow, open hashing, closed hashing, Issues in Hashing, Collision Resolution Strategies: Linear Probing, Quadratic Probing, Double Hashing, Open addressing and Chaining Representation of Symbol Tables- Static Tree Table and Dynamic Tree Table.	6
V	**Multiway trees and Heap** Multiway search tree, B+ Tree, Heap basic concepts, Realization of Heap, Heap as an Abstract Data Type, Heap implementation, Heap Sort, Heap as a Priority Queue.	6
VI	**File Organization** Files: concept, need, primitive operations. Sequential file organization-concept and primitive operations, Direct Access File- Concepts and Primitive operations, Indexed sequential file organization-concept, types of indices, structure of index sequential file, Pointer to file structure and basic operations on file.	6

List of Practical Assignments:
1. Construct an Expression Tree from postfix and prefix expression. Perform recursive and non-recursive In-order, pre-order and post-order traversals.
2. Implement binary search tree and perform following operations: a) Insert (Handle insertion of duplicate entry) b) Delete c) Search d) Display tree (Traversal) e) Display - Depth of tree f) Display - Mirror image g) Create a copy h) Display all parent nodes with their child nodes i) Display leaf nodes j) Display tree level wise
3. Implement In-order Threaded Binary Tree and traverse it in In-order, Pre-order and Post Order.
4. Represent a graph of your college campus using adjacency list /adjacency matrix. Nodes should represent the various departments/institutes and links should represent the distance between them.
5. Use Graph representation as adjacency list /adjacency matrix and find minimum spanning tree a) Using Kruskal's algorithm. b) Using Prim's algorithm.

6. Represent a graph of city using adjacency matrix /adjacency list. Nodes should represent the various landmarks and links should represent the distance between them. Find the shortest path using Dijkstra's algorithm from single source to all destination.
7. Consider telephone book database of N clients. Use Hash Table and Hash Functions to implement all the functions of a dictionary.
8. Implement various collision handling techniques to handle collisions in the Hash Table.
9. Implement Heap sort to sort given set of values using max or min heap.
10. Department maintains student's database. The file contains roll number, name, division and address. Write a program to create a sequential file to store and maintain student data. It should allow the user to add, delete information of student. Display information of a particular student. If the record of the student does not exist an appropriate message is displayed. If a student record is found it should display the student details:

Text Books :
1. E. Horowitz, S. Sahni, D. Mehta, "Fundamentals of Data Structures in C++", Galgotia Book Source, New Delhi, 1995, ISBN 16782928
2. Y. Langsam, M. Augenstin, A. Tannenbaum, "Data Structures using C and C++", 2nd Edition, Prentice Hall of India, 2002, ISBN-81-203-1177-9

Reference Books:
1. G. A.V, PAI , "Data Structures and Algorithms ", McGraw Hill, ISBN -13: 978-0-07-066726-6
2. M. Folk, B. Zoellick, G. Riccardi, "File Structure And Object Oriented Approach with C++", Pearson Education, 2002, ISBN 81 - 7808 - 131 - 8.
3. Sartaj Sahni. "Data Structures, Algorithms and Applications" in C++. Second edition, university press ISBN-13: 978-0072362268.

Online Resources:
1. https://nptel.ac.in/courses/106/102/106102064/
2. https://nptel.ac.in/courses/106/105/106105085
3. https:// nptel.ac.in/courses/106/106/106106127
4. https://www.ebookphp.com/advanced-data-structures-epub-pdf/
5. https://www.ebookphp.com/data-structures-and-algorithms-professional-edition-beginners-guide-epubpdf/

IT23212: Operating System

Course Code : IT23212	Course Credits: 04	Course type: PCC

Teaching Scheme			Evaluation Scheme					
TH	PR	TUT	ACTIVITY	ISE	ESE	TW	PR	OR
3	02 hrs./ week	–	20	20	70	20	20	-

Prerequisite Course Mapping:
1.   Basics of Computer Programming

Future Course Mapping:
1.   Ethical Hacking

Importance of Course: Students will learn what task OS is doing

Course Objectives:
1. Understand the basic concept and functions of the operating system
2. Understand the concept of Process and Thread management including scheduling synchronization, and deadlocks
3. Analyze the memory management techniques
4. Understand I/O Management and File System
5. Apply the Protection and Security to OS

Course Outcomes:
1. Understanding the role of Modern Operating Systems
2. **Apply** the concepts of process and thread scheduling
3. **Apply** the concept of process synchronization, mutual exclusion and the deadlock
4. **Understand** and apply the concepts of various memory management techniques
5. **Make** use of concept of I/O management and File system.
6. **Understand** Important of System software

Unit No.	Syllabus	Teaching Hours
1	**Introduction** Operating System Objectives, The Evolution of Operating System, OS Design Considerations for Multiprocessor and Multicore OS, Architectures of Operating System: Monolithic, Microkernel, Exokernel, Introduction to Linux OS, Basic Shell Commands, Shell Scripting using BASH.	6
	Process Management	

2	Process Concept, Process States, Process Control Block, Process Description Threads: Process and Threads, Basic types of threads, Multithreading, Thread Programming using thread library APIs Process Scheduling: Types of Scheduling, Scheduling Criteria, Scheduling Algorithms: First-Come First- Served, Shortest-Job-First, Priority, Round Robin, Case Study - Linux Scheduling	6
3	**Process Synchronization** Principles of Concurrency, Critical - Section Problem, Mutual Exclusion: Requirements, Operating System support - Semaphore and Mutex, Classical Synchronization Problems: Reader-Writer Problem, Producer - Consumer Problem, Real Life Problems, Inter-Process Communication: Pipes and Shared Memory Deadlock: Principles of Deadlock, Deadlock Characterization: Necessary Conditions, Resource - Allocation Graph, Methods for Handling Deadlock: Deadlock Prevention, Deadlock Avoidance, Deadlock Detection and Recovery, Case Study: Dining Philosopher Problem	6
4	**Memory Management** Memory Management Requirements, Memory Partitioning: Fixed Partitioning, Dynamic Partitioning, Buddy System, Paging, Segmentation, Virtual Memory: Demand Paging, Page Replacement, Thrashing, Case Study: Linux Operating System	6
5	**Input / Output and File Management** I/O Management: I/O Devices, Organization of the I/O Function, I/O Buffering, Secondary Storage Management: Disk Structure, Disk Scheduling File Management: Overview-Files and File Systems, File structure. File Organization and Access, File Directories, File Sharing, Case Study: Linux File System, Android File System	6
6	**Protection and Security** Goals of protection, Domain of protection, Access matrix, Implementation of access matrix, Revocation of access rights, Security problems, Authentication, Program threats, System threats, Threat monitoring	6

Practical Assignments		
Assign No	Title	
1	Study of Basic Linux Commands: echo, ls, read, cat, touch, test, loops, arithmetic comparison, conditional loops, grep, sed find, diff, tac and Bash scripting	
2	Create student datbse using Shell script	

3	Write a shell script to find factorial of number and check greater number between two numbers
4	Implement the C program in which the main function accepts the integers to be sorted. Main function uses the FORK system call to create a new process called a child process. Parent process sorts the integers using a sorting algorithm and waits for the child process using WAIT system call to sort the integers using any sorting algorithm. Also demonstrate zombie and orphan states.
5	Implement the C program to simulate any 2 CPU Scheduling Algorithms FCFS,Round Robin and Priority Scehduling(1 preemptive and 1 non- preemptive) with different arrival times.
6	Implement the C program for Producer Consumer problem using counting semaphores and mutex/binary semaphore.
7	Implement the C program for Reader-Writer problem with reader priority.
8	Implement the C program for Deadlock Avoidance Algorithm: Bankers Algorithm.
9	Implement the C program for any one Page Replacement Algorithm with minimum three frames as an input.(LRU,Optimal)
10	Implement Full duplex communication between two independent processes. First process accepts sentences and writes on the first pipe to be read by the second process. Second process counts number of characters, number of words and number of lines in accepted sentences, and writes the contents on second pipe to be read by first process and displays on standard output

Text Books :
1. William Stallings, Operating System: Internals and Design Principles, Prentice Hall, 8th Edition,2014, ISBN-10: 0133805913 • ISBN-13: 9780133805918
2. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, Operating System Concepts, John Wiley & Sons ,Inc., 9th Edition,2012, ISBN 978-1-118-06333-0
3. Arnold Robbins, Nelson H. F. Beebe, Classic Shell Scripting, O'Reilly Media, Inc., 2005, ISBN 9780596005955

Reference Books:
1. Tom Adelstein and Bill Lubanovic, Linux System Administration, O'Reilly Media, ISBN-10: 0596009526, ISBN-13: 978-0596009526.
2. Harvey M. Deitel, Operating Systems, Prentice Hall, ISBN-10: 0131828274, ISBN-13: 978-0131828278.
3. Thomas W. Doeppner, Operating System in depth: Design and Programming, WILEY, ISBN: 978- 0-471-68723-8
4. Mendel Cooper, Advanced Shell Scripting, Linux Documentation Project.
5. Andrew S. Tanenbaum & Herbert Bos, Modern Operating System, Pearson, ISBN-13: 9780133592221, 4th Edition.

Online Resources:
1.NPTEL Course "**Introduction to OS**"
https://onlinecourses.nptel.ac.in/noc21_cs72/preview

IT23213: Microprocessor & Microcontroller

Course Code : IT23213	Course Credits: 04	Course type: PCC

Teaching Scheme			Evaluation Scheme					
TH	PR	TUT	ACTIVITY	ISE	ESE	TW	PR	OR
3	2	-	20	20	70	20	20	-

Prerequisite Course Mapping:
1. Students are expected to have a good understanding of Logic Design and Computer Organization.

Future Course Mapping:
1. Microprocessor & Microcontroller laboratory

Importance of Course:
The microprocessors and microcontrollers are playing an important role in a wide range of engineering applications. Ultimately, microcontrollers and microprocessors are different ways of organizing and optimizing a computing system based on a CPU. Understanding microprocessor and Microcontroller design methodology as well as programming development will help to solve practical engineering problems.

Course Outcomes: On completion of the course, the learner will be able to

1. Apprehend the architecture and memory organization of the 80386 microprocessor
2. Understand memory management and multitasking of 80386 microprocessor
3. Apprehend the architecture and memory organization of the PIC 18 microcontroller
4. Implement embedded C programming for PIC 18.
5. Use concepts of timers and interrupts of PIC 18.
6. Demonstrate real life applications using PIC 18.

Course Outcomes:
1. Understand the object-oriented concepts.
2. Identify classes, objects, methods, and handle object creation, initialization, and Destruction to model real-world problems.
3. Identify relationships among objects using inheritance and polymorphism principles.
4. Understand the exception handling
5. Understand the generic programming and collection classes
6. Understand the file handling.

UNIT No.	Syllabus	Teaching Hours

I	**Introduction to 80386** History of Intel processors, Features of 80386DX, Architecture of 80386DX, 80386 Register Set, 80386 Real-Mode memory segmentation, Data types used in real mode, Instruction format.		6
II	**Bus Cycles & System Architecture** Processor state after reset, functional Pin Diagram, functionality of various pins, Memory organization (Memory Banks), 16/32 Bit Data Transfer Mechanism, Bus cycles of 80386, non-pipelined read/write cycle, pipelined read/write cycles with timing diagram.		6
III	**PIC Microcontroller Architecture** **Introduction:** introduction to microcontroller, Brief history of microcontrollers, Difference between microprocessor and microcontroller. **PIC18FXXX**: Features and architecture, comparison of PIC 18 series microcontrollers; PIC18F458/452 Pin out connection, Registers of PIC18F, **Program and data memory organization:** The Program Counter and Programmable ROM space in the PIC, File register and Access bank, Bank switching in PIC18; Oscillator configurations, Reset operations, Brownout reset, Watchdog timer.		6
IV	**PIC I/O Ports and Timer** I/O Port: I/O Port structure with programming: I/O Port structure, I/O Port programming, I/O Bit manipulation Programming. Timer/Counter: Registers used for Timer/Counter operation, Delay calculations, Programming of Timers using Embedded C.		6
V	**PIC Interrupts & Interfacing-I** PIC Interrupts: Interrupt Vs Polling, IVT, Steps in executing interrupt, Sources of interrupts; Enabling and disabling interrupts, Interrupt registers, Priority of interrupts, Programming of: Timer using interrupts, External hardware interrupts, Serial communication interrupt; Interfacing of LED, Interfacing 16X2 LCD (8 bits) , Interfacing Relay & Buzzer.		6
VI	**PIC Interfacing-II** CCP modes: Capture, Compare and PWM generation; DC Motor speed control with CCP, Basics of Serial communication protocols: Study of RS232, I2C, SPI, UART. Interfacing of ADC 0808 with PIC, Temperature sensor interfacing using ADC and I2C with PIC, Interfacing of RTC (DS1306) using I2C with PIC, Interfacing of EEPROM using SPI with PIC.		6
Total Teaching Hours			**36**

1) Write an embedded C program to sort array elements and show the sorted result on port or in memory register.
2) Write an Embedded C program to interface PIC 18FXXX with LED & blinking it using specified delay.
3) Write an Embedded C program for Timer programming ISR based buzzer on/off.
4) Write an Embedded C program for External interrupt input switch press, output at relay.
5) Write an Embedded C program for LCD interfacing with PIC 18FXXX.
6) Write an Embedded C program for Generating PWM signal for servo motor/DC motor.
7) Study of Arduino board and its pins, sensors and components.
8) Write a simple program using Open source prototype platform like Raspberry-Pi/Beagle board/Arduino for digital read/write using LED and switch Analog read/write using sensor and actuators.

Text Books:
1) 1. Douglas Hall, " Microprocessor and Interfacing", McGraw Hill, Revised 2nd Edition, 2006ISBN 0-07-100462-9
2) 2. James Turley, "Advanced 80386 Programming Techniques", McGraw Hill Education
3) 3. A. Ray, K. Bhurchandi , "Advanced Microprocessors & peripherals: Arch, programming and interfacing ", Tata McGraw Hill , 2004 ISBN 0-07-463841-6.
4) 4. Muhammad Ali Mazidi , Danny Causey, RolinMcKinlay, "PIC Microcontroller and Embedded Systems: Using Assembly and C for PIC18", 4th Edition by,Pearson international edition

Reference Books:
1) Joshi, "Processor Architecture and Interfacing", Wiley, ISBN-9788126545605
2) Peatman, John B, "Design with PIC Microcontroller", Pearson Education PTE
3) 3. Ramesh Gaonkar, "Fundamentals of Microcontrollers and Applications In Embedded Systems(with the PIC18 Microcontroller Family)" Thomson/Delmar Learning; 1 edition (January 8, 2007), ISBN:978-1401879143.
4) 4. Microchip's PIC18FXXX Data Sheet .

IT23214: Software Engineering

Course Code : IT23214	Course Credits: 03	Course type: PCC
Teaching Scheme	**Evaluation Scheme**	

TH	PR	TUT	ACTIVITY	ISE	ESE	TW	PR	OR
3	-	-	20	20	70			-

Prerequisite Course Mapping:
1. Fundamentals of Programming Languages

Future Course Mapping:
1.This course helps to design and modelling software projects
2.This course will provide an in-depth analysis for software Engineer to develop software projects in IT industries

Importance of Course:
1. This course equips students with a solid foundation of technical knowledge and problem-solving skills.
2. Software engineering concepts used to manage the complexity of their software development projects.
3. It empowers them to identify improvement areas and find new ways to reduce costs.

Course Objectives:
1. To learn the principles of Software Engineering.
2. To learn and understand methods of capturing, specifying, visualizing and analyzing software requirements.
3. To know design principles to software project development.
4. To learn basics of IT project management.
5. To understand software quality attributes and testing principles.
6. To introduce formal methods and recent trends in Software Engineering.

Course Outcomes:
1. Unit1: Classify various software application domains
2. Unit2: Analyze software requirements by using various modelling techniques
3. Unit3: Translate the requirement models into design models
4. Unit4: Apply planning and estimation to project
5. Unit5: Use quality attributes and testing principles in software development life cycle
6. Unit6: Discuss recent trends in Software engineering by using CASE and agile tools.

Syllabus

UNIT No.	Syllabus	Teaching Hours
I	**Software Engineering Fundamentals:** Nature of Software, Software Engineering Practice, Software Process, Software Myths. **Process Models :** A Generic Process Model, Linear Sequential Development Model, Iterative Development Model, The incremental Development Model **Agile software development:** Agile manifesto, agility principles, Introduction to Extreme programming and Scrum.	06 hrs
II	**Requirements Engineering:** User and system requirements, Functional and non-functional requirements, requirements engineering (elicitation, specification, validation, negotiation) prioritizing requirements (Kano diagram), Software Requirements Specification (SRS) **Requirements Analysis:** Analysis Model, data modelling, Introduction to	06 hrs

		scenario based modelling, class based modelling, Flow oriented modelling, behavioural modelling	
III		**Design Engineering:** Design Process & quality, Design Concepts, design Model **Architectural Design** :Design Decisions, Views, Patterns, Application Architectures, **Component level Design:** component, Designing, class based components, conducting component-level design **User Interface Design:** The golden rules, Interface Design steps& Analysis, Design Evaluation	06 hrs
IV		**Project Planning:** Project initiation, Planning Scope Management, Creating the Work Breakdown Structure, **scheduling:** Importance of Project Schedules, Schedule development using Gantt Charts, PERT and CPM **Project Management:** The Management Spectrum, The W5HH Principle, Software Measurement: size & function-oriented metrics(FP & LOC), **Project Estimation:** Software Project Estimation, Decomposition Techniques, Cost Estimation Tools and Techniques, Typical Problems with IT Cost Estimates	06 hrs
V		**Quality Concepts:** Quality, software quality, Quality Metrics, software quality dilemma **Software Testing:** Introduction to Software Testing, principles of Testing, Test plan, Test case, Types of Testing, Verification & Validation, Testing strategies, Defect Management, Defect Life Cycle, Bug Reporting, debugging	06 hrs
VI		**Recent Trends in Software Engineering :** SCM, Risk Management, software reuse, global software development challenges **CASE :** Taxonomy, tools ,components of CASE, categories (upper, lower and integrated CASE tools), Introduction to agile tools Jira, Kanban	06 hrs

Text Books:

1.Roger Pressman, "Software Engineering:A Practitioner's Approach", McGraw Hill,ISBN 0-07-337597-7

2. Ian Sommerville, "Software Engineering",Addison and Wesley, ISBN 0-13-703515-2

Reference Books:

1.Joseph Phillips, "IT Project Management-On Track From start to Finish", Tata Mc Graw-Hill,ISBN13:978-0-07106727-0,ISBN-10:0-07-106727-2

2. Pankaj Jalote, "Software Engineering: A Precise Approach",Wiley India, ISBN: 9788-1265-2311-5

3. Marchewka, "Information Technology Project Management",Willey India, ISBN: 9788-1265-4394-6

4. Rajib Mall, "Fundamentals of Software Engineering",Prentice Hall India, ISBN-13:9788-1203-4898-1

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc20_cs68/preview
2. http://vlabs.iitkgp.ernet.in/se/

Dr. T.V.Bhandare
HoD – IT

Dr. S.M. Bhosle
Dean Academics

Dr. R.S. Bichkar
Principal