Bansilal Ramnath Agarwal Charitable Trust's

Vishwakarma Institute of Technology

(*An Autonomous Institute affiliated to Savitribai Phule Pune University*)

Structure & Syllabus of

Second Year B.Tech.

(Instrumentation and Control Engineering)

Pattern 'B20'

Effective from Academic Year 2020-21

Prepared by: - Board of Studies in Instrumentation & Control Engineering

Approved by: - Academic Board, Vishwakarma Institute of Technology, Pune

Signed by

Chairman – BOS **Chairman – Academic Board**

Contents

Vision statement of Institute

To be globally acclaimed Institute in Technical Education and Research for holistic Socio-economic development

Mission statement of Institute

- To endure that 100% students are employable in Industry, Higher studies, Become Entrepreneurs, Civil/Defense Services / Government Jobs and other areas like Sports and Theatre.
- To strengthen Academic Practices in terms of Curriculum, Pedagogy, Assessment and Faculty Competence.
- Promote Research Culture amongst Students and Faculty through Projects and Consultancy.
- To make students Socially Responsible Citizen.

Core Values

- Faculty Centric Initiatives
- Academic Practices
- Research Culture
- Use of Technology for Social and National Development

Vision statement of Department

To be recognized as a leading contributor in imparting technical education and research in Instrumentation & Control engineering for development of the society.

Mission statement of Department

- To deliver knowledge of Instrumentation and Control Engineering by strengthening involvement of Research institutions and industries in academics
- To build conducive environment for advanced learning through participation of faculty and students in collaborative research, consultancy projects, student exchange programs and internships
- To develop competent Engineers with entrepreneurial skills to address socio-economic needs.

Program Educational Objectives (PEO)

Programme: B. Tech. (Instrumentation and Control Engineering)

The Graduates would demonstrate

1. Core competency in Instrumentation and Control Engineering to cater to the industry and research needs.
2. Multi-disciplinary skills, team spirit and leadership qualities with professional ethics, to excel in professional career and/or higher studies.
3. Preparedness to learn and apply contemporary technologies for addressing impending challenges for the benefit of organization/society.
4. Knowledge of recommended standards and practices to design and implement automation solutions.

Program Outcomes

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 modeling 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 teamwork:** 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.

Program Specific Outcomes (PSOs)

Graduates shall have the ability to:

1. Evaluate the performance of suitable sensors / Process components/ Electronic / Electrical components for building complete automation systems.
2. Analyze real-world engineering problems in the area of Instrumentation and Control.
3. Design or Develop measurement / electronic / embedded and control systems with computational algorithms to provide practical solutions to multidisciplinary engineering problems.

Vishwakarma Institute of Technology
Title : Course Structure

Issue 01 : Rev No. 1 : Dt. 01/07/18
FF No. 653

S.Y. B.Tech - Instrumentation and Control Engineering Structure for Pattern B-20, Module-4 with effect from Semester-2 of Academic Year 2020-21

Course Type	Course Code	Course Name	Teaching Learning Scheme (Hrs./Week)				Credits	Assessment Scheme (100 mark scale)							
			Th	Lab	Tut	Total		In Semester Assessment (65)					End Semester Assessment (35)		Total
								HA	Lab	Seminar	GD/PPT	MSE	ESE	Viva	100
S1	IC2202	Data Structures	3	2	1	6	5	10	20	10	10	15	15	20	100
S2	IC2204	Computer Architecture and Operating System	3	2	1	6	5	10	20	10	10	15	15	20	100
S3	IC2206	Signal and Image Processing	3	2	1	6	5	10	20	10	10	15	15	20	100
S4	IC2208	Engineering Design - II	0	2	0	2	1		50				50		100
S5	IC2210	Software Design - II	0	2	0	2	1		50				50		100
S6	IC2212	Software Development Project - II	0	6	0	6	3		50				50		100
S7	IC2214	Engineering Design and Innovation - IV	0	8	0	8	4		50				50		100
		Total	**9**	**24**	**3**	**36**	**24**								

Note : Courses S4 to S7 are independent of module and offered in Semester-2

SEMESTER II

FF No. : 654

IC2202 :: DATA STRUCTURES

Course Prerequisites: Basic understating of C/C++ programming

Course Objectives:

1. To impart the basic concepts of data structures and algorithms.
2. To understand concepts about searching and sorting techniques.
3. To construct and implement various data structures and abstract data types including lists, stacks, queues, trees, and graphs.
4. To make understand about writing algorithms and step by step approach in solving problems with the help of fundamental data structures.
5. To emphasize the importance of data structures in developing and implementing efficient algorithms

Credits: 5

Teaching Scheme Theory: 3 Hours/Week
Tut : 1 Hour/Week
Lab : 2 Hours/Week

Course Relevance:

In the current scenario there is huge demand in the software industry for a skilled personnel. Data Structures is a basic fundamental course for any student who wishes to make a career in the field of Software industry. Almost every industry working in the software sector be it in the development, service, banking and finance, Data analysis and allied looks for an engineer who has basic programming and analysis skills. Data structures gives the students the insights of the working of program and file handling. The course content is included in such a way that the beginner can easily understand the course. The course has hands-on sessions so as to make the learner understand the concepts clearly. Once a student successfully learns the concepts of DS s/he can certainly implement the same in real world engineering applications such as Billing management systems, Database handling, File handling and operations, NLP implementation, Shortest path finding problem solving are to name a few.

Moreover the Students who are aspiring to pursue a master's degree in the field of Engineering are also expected to have a deep understanding of concepts of Data Structures.

This course will develop a core competency in the field of Software technology.

SECTION-1: [IC2202_CO1, IC2202_CO2, IC2202_CO3]

Fundamentals of Data Structures:
Introduction to DS, Types: Primitive, Non primitive, Linear, Nonlinear, Static, Dynamic structures, Pointers in C, Single and Multidimensional arrays: Memory representation and indexing, operations on multidimensional arrays. Time & Space Complexity Analysis.

Sorting Techniques:
Understanding of Selection, Bubble, Insertion, Merge, Quick, Heap sort techniques. Understanding Time and Space complexities of the algorithms

Searching: Understanding Linear and Binary Search algorithm and Time and Space complexities

Linked Lists: Singly Linked Lists, Doubly linked Lists, Circular linked lists, Applications: Stack & Queue using linked list, Polynomial Manipulation using linked list.

Stack: representation using array, Applications of stack: Recursion, Validity of parentheses, Expression conversions and evaluations etc.

Queue: representation using array, Types of queue, Applications of Queue: Job Scheduling, Josephus problem etc.

SECTION-1I: [IC2202_CO4, IC2202_CO5,IC2202_CO6]

Trees: Basic terminology, representation using array and linked list, Tree Traversals: Recursive And Non recursive, counting no of Nodes etc., Construction of binary tree from traversals, Binary Search trees(BST): Insertion, deletion of a node from BST. Threaded Binary tree (TBT): Creation and traversals on TBT, Expression Trees, Gaming Trees

Advanced Trees: Introduction, AVL tree, R-B tree, B tree and B+ tree.

Graphs: Terminology and representation using Adjacency Matrix and Adjacency Lists, Graph Traversals and Application: BFS and DFS, Connected graph, Bipartite Graph, Detecting Cycle in graph. Minimum Spanning tree: Prims and Kruskal's Algorithm, Shortest Path Algorithms, Union Find.

Hashing: Hashing techniques, Hash table, Hash functions. Collision handling and Collision resolution techniques.

List of Tutorials: (Any Three)

1. Sorting Techniques: Insertion, Merge sort, Bubble, Shell Sort, Radix Sort.
2. Searching Techniques: Ternary Search, Fibonacci Search.
3. Problem solving using stack (Maze problem, Tower of Hanoi).
4. Expression conversion like infix to prefix and postfix and vice versa.
5. Priority Queues and Job Scheduling Algorithm.
6. Generalized Linked Lists.
7. Threaded Binary tree and Stack less Traversals using TBT.
8. B and B+ Tree.
9. Applications of Graph in Network problems.
10. Design of Hashing Functions and Collision Resolution techniques.
11. Cuckoo Hashing**.**

List of Practicals: (Any Six)

1. Assignment based on Sorting and Searching.
2. Assignment based on Stack Application (Expression conversion etc.)
3. Assignment based on Queue Application (Job scheduling, resources allocation etc.)
4. Assignment based on linked list.
5. Assignment based on BST operations(Create, Insert, Delete and Traversals)
6. Assignment based on various operations on Binary Tree (Mirror image, Height, Leaf node display, Level wise display etc.)
7. Assignment based on AVL and R-B tree.
8. Assignment based on DFS and BFS
9. Assignment based on MST using Prim's and Kruskals Algorithm.
10. Assignment based on Finding shortest path in given Graph.
11. Assignment based on Hashing.

List of Projects:

1. Billing management system
2. Student database creation using Linked list
3. Expression tree
4. Sudoku solver
5. Dictionary using Tree
6. Movie ticket booking system
7. Maze runner problem solver
8. Calendar application using structures
9. Solving Tower of Hanoi puzzle
10. Finding shortest path in a given map
11. Finding Nearest Neighbors.
12. Calendar Application using File handling.
13. Word Completion Using Tire.
14. Bloom Filters.
15. Different Management Systems.
16. Scheduling Applications and Simulation.
17. Efficient Storage and Data Retrieval Systems.
18. Different Gaming Application.

List of Course Seminar Topics:

1. Implementation of Sorting techniques
2. Implementation of Searching techniques
3. Expression conversion using Stack and Queue
4. Graph colouring
5. Bipartite graphs
6. Data management system using DS
7. File handling and memory optimization using DS
8. Solving logical puzzles using DS
9. Language dictionary creating using Linked list
10. Use of Trees in Natural Language Processing

List of Course Group Discussion Topics:

1. In place sorting VS Not in place sorting
2. Selecting search algorithm
3. Selecting a Graph or a tree data structure
4. Graph traversal using BFS and DFS what is the best choice?
5. Is Finding the shortest path given a graph possible? How?
6. Is it necessary to find MST? Which technique is better?
7. Recursion makes coding easier. Is it really so?
8. How does Boggle help in the Mazing problem?
9. Finding a winning position using Josephus' concept is trivial.
10. Linear Vs Non-Linear Data Structures with applications
11. How do you parse expressions using trees?
12. Contiguous memory allocation VS Non contiguous memory allocation
13. How linear probing is useful in Hashing?
14. Adjacency list or Adjacency matrix which is a better choice?
15. Linked lists are popular among all data structures. Is it right?

List of Home Assignments:

Design:

1.Shortest path finding algorithms
2.Formation of trees for various applications
3.Forming minimum spanning tree
4.Implementation of 3-d array
5.Implemenation of Best first search algorithm

Case Study:

1.Hashing and its use in DS
2.Bidirectional search algorithm
3.Comparison of path finding algorithm
4.Solving TSP
5.Process scheduling

Blog

1. Reducing time complexities of algorithm
2.Greedy algorithm and its use
3.Cascading Searching techniques
4. Solving Mazing problem
5.Solving Recurrence relations using DS

Surveys
1.A survey on Blind searching techniques
2.Use of Recursion and Backtracking
3.Survey of MST algorithms
4. Survey of Shortest path finding algorithms
5.Applications of Stack and Queue

Suggest an assessment Scheme:
Course Assessment: Total : 100 mks
1. Mid Semester Examination : 15 marks (Total : 30 marks based on Section I)
30 marks converted to 15
2. End Semester Examination: 15 marks (Total : 30 marks based on Section II)
30 marks converted to 15
3. Lab Assessment: Lab Assignments - 20 marks (10 assignments-10 marks each)
(*100 marks converted to 20)*
4. Course Project : 20 marks (*100 marks converted to 20)*
5. Home Assignment : 10 mks (Total : 100 marks: Case study, Design work, Survey, Blog)
100 marks converted to 10
6. Viva : 20 mks (at the end of semester); *100 marks converted to 20*

Text Books:

1. "Fundamentals of Data Structures in C", E. Horwitz , S. Sahani, Anderson-Freed, Second Edition, Universities Press
2. "Problem solving using Data Structures & Algorithms using C", Hemant Jain, First edition

Reference Books:

1. Data structures using C and C++", Y. Langsam, M.J. Augenstein, A.M.Tenenbaum, Pearson Education, Second Edition

Moocs Links and additional reading material:
1.nptel.ac.in/courses/106/102/106102064
2.nptel.ac.in/courses/106/105/10610508

Course Outcomes: The student will be able to –
1. IC2202_CO1: Interpret and diagnose the properties of data structures with their memory representations [1] (PO 1,2,3,12, PSO 3)
2. IC2202_CO2: Comprehend various sorting and searching algorithms. [2] (PO 1, 2,3,4,12, PSO 3)
3. IC2202_CO3: Use linear data structures like stacks, queues etc. with their applications [2] (PO 1, 2,3,12, PSO 3)
4. IC2202_CO4: Handle operations like searching, insertion, deletion, traversing mechanism etc. on tree. [3] (PO 1, 2,3,4,12, PSO 3)
5. IC2202_CO5: Demonstrate the use of binary tree traversals and to perform various operations on nonlinear data structures. [3] (PO 1, 2,3,4,12, PSO 3)
6. IC2202_CO6: Implement the graph data structures to solve engineering problems. [4] (PO 1, 2,3,4,12, PSO 3)

CO PO Map:

CO/PO	PO:1	PO:2	PO:3	PO:4	PO:5	PO:6	PO:7	PO:8	PO:9	PO:10	PO:11	PO:12	PSO:1	PSO:2	PSO:3
CO:1	1	1	1	0	0	0	0	0	0	0	0	2	0	2	0
CO:2	0	2	2	2	2	0	0	0	0	0	0	0	2	0	2
CO:3	0	0	1	2	2	0	0	0	0	0	0	0	0	2	0
CO:4	2	0	0	1	2	2	2	0	0	0	0	0	0	0	2
CO:5	0	2	0	0	1	2	2	2	0	0	0	0	0	0	0
CO:6	2	0	2	0	0	1	2	2	2	0	0	0	0	0	0

CO attainment levels

CO No	IC2202_CO1	IC2202_CO2	IC2202_CO3	IC2202_CO4	IC2202_CO5	IC2202_CO6
Attainment level	1	2	2	3	3	4

Future Courses Mapping:

Data Analytics, Data Science, OOPS, Artificial Intelligence, Machine learning, Software Engineering

Job Mapping:

DS is a fundamental course for the students who are aspiring to make a career in the field of Software engineering. Once this course is learned students can apply for a job as a Software developer, Software testing, Data Analyst etc. The course is also helpful for students who are planning to have their own start-ups in the area of Software development.

FF No. : 654

IC2204 :: COMPUTER ARCHITECTURE AND OPERATING SYSTEM

Course Prerequisites: Basics of computer system, data structures and programming languages

Course Objectives:

1. To illustrate the structure, function, characteristics and performance parameters of a computer system.
2. To explore several computer architectures.
3. To discuss memory organization in computer system
4. To understand the basic concepts and functions of the operating system.
5. To gain knowledge of process synchronization, its mechanism and CPU scheduling
6. To get familiar with deadlock and memory management techniques as a function of the operating system.

Credits: 5

Teaching Scheme Theory: 3 Hours/Week
Tut : 1 Hour/Week
Lab : 2 Hours/Week

Course Relevance:

This course focuses on CISC and RISC computer architecture. The Operating System acts as a platform for information exchange between your computer's hardware and the applications running on it. A computer software/hardware architect is deeply involved in the development and design of new software or hardware.

SECTION-1
Topics and Contents **Introduction:** Evolution of Computer Systems, Basic Operation of a Computer, Memory Addressing and Languages, Software and Architecture Types **CISC:** Architecture of 8086, Instruction types, instruction format, instruction cycle, Addressing Modes, Assembly Language Programming of 8086, **RISC:** Architecture, Instruction set, Pipelining, Programming and Application **Measuring CPU performance**: Choice of benchmarks, summarizing performance results, Amdahl's law **Control Unit:** Single Bus CPU organization, register transfers, performing an arithmetic/ logic operation, fetching a word from memory, storing a word in memory, Execution of a complete instruction. Micro-operations, Hardwired Control, Micro-programmed Control: Microinstructions **Memory System:** Need of memory system, Hierarchical memory system, Characteristics, Size, Access time, read cycle time and address space, Processor memory interaction, Static and Dynamic ram, Memory interfacing and addressing, Memory hierarchy design, Cache memory: Cache size vs block size, Mapping functions.

SECTION-1I

Overview of Operating System: What is OS?, Interaction of OS and hardware, Goals of OS, Basic functions of OS, OS Services, System Calls, Types of system calls, Types of OS: Batch, Multiprogramming, Time sharing, Parallel, Distributed & Real-time OS

Process management: Process Concept, Process States: 2, 5, 7 state models, Process Description, Process Control, Multithreading models, Thread implementations – user level and kernel level threads, Concurrency: Issues with concurrency, Principles of Concurrency, Mutual Exclusion: OS/Programming Language Support: Semaphores, Mutex , Classical Process Synchronization problems, Uniprocessor Scheduling, Scheduling Algorithms: FCFS, SJF, RR, Priority.

Deadlock: Principles of deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Deadlock Recovery, **Memory Management:** Requirements, Memory Partitioning, Fragmentation, Placement Strategies: First Fit, Best Fit, Next Fit and Worst Fit, Paging, Segmentation, Address Translation, Virtual Memory, VM with Paging, Page Table Structure, Translation Lookaside Buffer, Page Size, VM with Segmentation, Page Replacement Policies: FIFO, LRU, Optimal

List of Tutorials: (Any Three)

1) Instructions encoding.
2) Performance parameter (Amdahl's law)
3) Performance through pipelining.
4) Control Unit: Micro-operation and Micro-instruction.
5) Cache mapping functions.
6) Draw the Gantt charts and compute the finish time, turnaround time and waiting time for the following algorithms:
   a) First come First serve b) Shortest Job First (Preemptive and Non preemptive)
   c) Priority (Preemptive and Non preemptive) d) Round robin
7) Check whether the given system is in a safe state or not using Banker's Deadlock Avoidance algorithm.
8) Check whether the given system is in a deadlock state or not using the Deadlock Detection algorithm.
9) Using the following placement algorithm, check whether memory can be allocated to a given process or not.
   a) First fit b) Best fit c) Worst fit d) Next fit
10) Calculate the number of page faults for a reference string for the following page replacement algorithms:
   a) FIFO b) LRU c) Optimal

List of Practicals: (Any Six)

1) Write an ALP to perform arithmetic operations.
2) Write an ALP using an array.
3) Write an ALP using stack memory.
4) RISC Programming
5) Execution of Basic and Advanced Linux commands
6) Write a shell script program.
7) Write a program demonstrating use of different system calls.
8) Implementation of Classical problems using Threads and Mutex/Semaphore.
9) Write a program to compute the finish time, turnaround time and waiting time for the following algorithms: a) First come First serve b) Shortest Job First (Preemptive and Non preemptive) c) Priority (Preemptive and Non preemptive) d) Round robin
10) Write a program to check whether given system is in safe state or not using Banker's Deadlock Avoidance algorithm
11) Write a program for following placement algorithm check whether memory can be allocated to given process or not by using following methods: a) First fit b) Best fit c) Worst fit d) Next fit
12) Write a program to calculate the number of page faults for a reference string for the following page replacement algorithms: a) FIFO b) LRU c) Optimal

List of Projects:

1. Linux based application using Shell Scripting
2. Design and implementation of a Multiprogramming Operating System: Stage I
   i. CPU/ Machine Simulation
   ii. Supervisor Call through interrupt
3. Design and implementation of a Multiprogramming Operating System: Stage II
   i. Paging
   ii. Error Handling
   iii. Interrupt Generation and Servicing
   iv. Process Data Structure
4. Design and implementation of a Multiprogramming Operating System: Stage III
   i. Multiprogramming
   ii. Virtual Memory
   iii. Process Scheduling and Synchronization
   iv. Inter-Process Communication
   v. I/O Handling, Spooling and Buffering
5. Design and implementation of a Multiprogramming Operating System for arithmetic and logical operations: Stage I
   i. CPU/ Machine Simulation
   ii. Supervisor Call through interrupt
6. Porting of Linux on Embedded Platform and basic I/O programming
7. Comparison of various processors using simulators.
8. RTOS Programming

9. Designing of CPU
10. Linux kernel programming
11. Parallel Computing using CUDA

List of Course Seminar Topics:

1. Pentium Processor - a complete architecture
2. Microprogram sequencing
3. Improvement of Performance Measurement of Processor: Memory Banking
4. GPU Architecture
5. Micro-Programmed Control Unit used in Recent Computer.
6. Parallel Computers
7. I/O processors
8. Effect of clock on CPU performance
9. Edge computing
10. In-Memory Computing
11. Computer Architectures for vision system
12. RISC -V architecture
13. Cyber Physical Systems
14. Cyber System Debugging
15. Neuromorphic computing
16. Quantum Computing
17. The Challenges of Building Inferencing Chips
18. Hardware accelerator in computer architecture

List of Course Group Discussion Topics:

1. OS Structures
2. System call Vs API
3. Classical process synchronization problems
4. Process Vs Threads
5. Interprocess Communication (IPC)
6. Real Time Scheduling
7. Disk Scheduling
8. Best OS for smartphones-Android, iOS, windows, blackberry
9. Shared and Distributed Memory microprocessor
10. Flynn's Taxonomy
11. Booting Process of different Operating Systems.

List of Home Assignments:

Design:

1. Report Generation using Shell Script and AWK
2. Library Management System using Shell
3. Inter Process Communication in Linux
4. Design any real time application using job scheduling
5. Design any application using Android
6. DRAM design
7. Embedded System design
8. Real Time System design

Case Study:

1. Intel I3
2. Intel I7
3. Microsoft Windows 10
4. Linux
5. Android
6. Raspberry PI
7. NVIDIA core
8. Supercomputer architecture

Blog

1. ARM Vs Intel
2. Protection and Security of OS
3. Comparative study of different mobile OS
4. Operating Systems for IoT Devices
5. Performance Measurement of CPU: Pipelining
6. ARM Microcontroller versions
7. Operating System Forensics
8. IOT Architecture

Surveys

1. Computer System Memory Management and Optimization Techniques
2. Multiprocessor organization
3. A Survey of Mobile OS
4. Analysis and Comparison of CPU Scheduling Algorithms
5. Malware Analysis, Tools and Techniques
6. Laptop Operating Systems
7. Desktop Operating Systems
8. Pipelining hazards
9. Elements of modern computers

Assessment Scheme:

1. Home Assignment: Design, Case study, Blog and Survey
2. MCQ
3. CVV
4. Seminar
5. Group Discussion
6. LAB-Lab expt and Course Project

Text Books:

1. William Stallings; "Computer Organization and Architecture: Designing for Performance"; 7th Edition; Pearson Prentice Hall Publication;ISBN 81-7758-9 93-8
2. C. Hamacher, V. Zvonko, S. Zaky; "Computer Organization"; 5th Edition; Tata McGraw Hill Publication; ISBN 007-120411-3
3. Douglas Hall; "Microprocessors and Interfacing";2nd Edition; Tata McGraw Hill Publications; ISBN 0-07-025742-6
4. Stalling William; "Operating Systems"; 6th Edition; Pearson Education; ISBN: 0-13-031999-6
5. Silberschatz A., Galvin P., Gagne G.;"Operating System Concepts" ; 9th Edition; John Wiley and Sons
6. John L. Hennessy, David A. Patterson; " Computer Architecture-A Quantitative Approach", 5th edition, Elsevier Publication
7. Andrew Sloss, Dominic Symes, Chris Wright; "ARM System Developer's guide Designing and optimizing system software"; Elsevier Publication

Reference Books:

1. Hwang and Briggs; "Computer Architecture and Parallel Processing"; Tata McGraw Hill Publication; ISBN 13: 9780070315563.
2. A. Tanenbaum; "Structured Computer Organization"; Prentice Hall Publication; ISBN 81 – 1553-7.
3. Silberschatz A., Galvin P., Gagne G; "Operating System Principles"; 7th Edition, John Wiley andSons.
4. Yashavant Kanetkar; "Unix Shell Programming"; 2nd Edition, BPB Publications
5. Sumitabha Das; "Unix Concepts and Applications"; 4th Edition, TMH.
6. Forouzan B. A., Gilberg R. F.; "Unix And Shell Programming"; 1st Edition, Australia Thomson Brooks Cole.

Moocs Links and additional reading material:

1. www.nptelvideos.in
2. https://www.udemy.com/
3. https://www.coursera.org/
4. https://swayam.gov.in/

Course Outcomes: The students are able to:

The student will be able to –

1) Illustrate the structure, function, characteristics and performance parameters of a computer system such as benchmarks, Amdahl's law, price and power.

2) Explore the knowledge of Computer Architectures such CISC and RISC

3) Discuss static, dynamic and cache memory in computer system

4) Understand the functions of a contemporary Operating system with respect to convenience, efficiency and the ability to evolve.

5) Apply various CPU scheduling algorithms and process synchronization mechanisms.

6) Identify the mechanisms to deal with Deadlock and primary memory management.

CO PO Map

CO/ PO	PO 1	PO2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	3	2	1	-	-	-	1	-	1	2	-	1	-	-	2
CO2	3	-	1	-	-	-	1	-	1	2	-	1	2	1	2
CO3	2	1	0	1	-	-	-	-	1	2	-	1	-	1	2
CO4	2	-	-	-	-	2	2	1	2	2	1	1	1	1	1
CO5	1	2	2	1	2	1	1	1	2	1	1	1	1	1	2
CO6	1	2	2	1	2	1	1	1	2	1	1	1	-	1	2

CO attainment levels

CO NO	CO1	CO2	CO3	CO4	CO5	CO6
Attainment level	2	3	2	2	3	3

Future Courses Mapping: Advance Computer Architecture, Advance Operating System, Unix Operating System, Linux programming, Distributed System/Computing, High Performance Computing, Embedded Systems, System Programming, Compiler
Job Mapping: Linux Administration, Kernel Developers, Application Developers, System programmer, Computer Architects, Cyber Security analyst, System administrator

FF No. : 654

IC2206 :: SIGNAL AND IMAGE PROCESSING

Course Prerequisites: mathematics, mathematical transforms, linear algebra

Course Objectives:

1. Upon completion of this course, students will be familiar with basic signal and image processing techniques for solving real problems.
2. This course will provide understanding to design, implement, and analysis digital system utilizing the signal and image processing techniques

Credits: 5 **Teaching Scheme Theory: 3** Hours/Week

Tut : 1 Hour/Week
Lab : 2 Hours/Week

Course Relevance:

Signal and Image processing course is the basis of machine learning, deep learning, artificial intelligence, robotics, automation, IOT, Industry 4.0

SECTION-1

Section 1 : Signal Processing

Signals and Systems

Signals – Continuous time and discrete time, time shift and time scale operations on signals, sampling, discrete time systems - memory-less systems, linear time invariant systems, causality, stability properties of linear time- invariant systems. Convolution of discrete time sequences, properties of convolution.

DFT and IDFT

Fourier transform, Discrete Fourier Transform (DFT): DFT and IDFT, Properties of DFT, Circular convolution, linear convolution using DFT and IDFT

FFT and IFFT

Fast Fourier Transform (FFT and IFFT) – Radix 2 - DIT and DIF algorithms.

FIR Filter Design

FIR filter design using windowing techniques. Low pass, High pass, Band Pass, Band stop filter design by windowing method

Analog Filter design and IIR Filter design

Analog filter design: Butterworth filters, Low pass Butterworth filter design. Digital IIR filter design: Bilinear transformation

SECTION-1I

Section 2: Image Processing

Digital Image Fundamentals

Introduction to Digital Image, Digital Image Processing System, Sampling and Quantization

Image Transformation – 2 -D DFT, 2-D FFT, DCT

Image Enhancement and Restoration –

Histogram Processing, Histogram equalization.

Spatial Filtering, Smoothing and Sharpening Filters, Median Filter

Image Segmentation

Segmentation based on Discontinuities (point, Line, Edge)

Image Edge detection using Robert, Sobel, Previtt masks, Canny's Edge Detector

List of Tutorials: (Any Three)

1) Sampling Theorem - Computation and Plotting of Continuous time and Discrete time signals
2) Computation of time shift and time scale operations on discrete signals
3) Computation of convolution
4) Computation of 1 -D DFT
5) Computation of FFT radix 2 – DIT and DIF algorithm
6) Designing FIR Filter using windowing technique
7) Designing low pass IIR Filter using BZT method
8) Computation of Geometric transforms on image – scale, rotate, reflect translate (considering 3*3 matrix)
9) Computation of 2 D – DFT (considering 3*3 matrix)
10) Computation of new image - smoothening using averaging filter

List of Practicals: (Any Six)

1. Plot of continuous time and discrete time sinusoidal signal – understanding the concept of Sampling Theorem
2. Time Shift and Time scale operations on Discrete signal
3. Linear Convolution
4. To perform Fast Fourier transforms using radix 2 DIT, DIF algorithms.
5. FIR filter design by using windowing technique.
6. IIR Filter Design by using Bilinear Transformations method.
7. Formation of image from a 8*8 matrix, plotting negative image 8*8 matrix
8. Geometric operations on a image scale, rotate, reflect translate
9. Plotting histogram of an image, studying the effect of varying pixel weight
10. Convolution of two images
11. 2 D - FFT of image
12. Computation of new image - smoothening using averaging filter

List of Projects Areas:

1. Audio signal filtering
2. Biomedical Signal analysis
3. Frequency spectrum analysis
4. Vibration Signal analysis
5. Speech recognition
6. Image filtering
7. Image recognition
8. Image Classification
9. Object detection using image processing
10. Image compression

List of Course Seminar Topics:

1. Signal Processing Applications in sensing and measurement
2. Signal Processing application in Robotics
3. Signal Processing applications in Automation
4. Signal Processing application in Industry
5. Signal Processing application in Space technology
6. Image Processing application in automation
7. Image processing application in robotics
8. Image processing application in Industry 4.0
9. Image processing application in Consumer Electronics
10. Signal processing application in consumer Electronics

List of Course Group Discussion Topics:

1. Sampling theorem, ADC, real time implementation effects, aliasing
2. Hardware requirement for DSP systems vs speed of operation of the system
3. Hardware implementation of signal processing in measurement system
4. Hardware implementation of image processing algorithms in biomedical equipment
5. Hardware implementation of signal and image processing in Industry 4.0
6. Implementation and Accuracy of signal processing algorithms for given application
7. Implementation and Accuracy of image processing algorithm for given application
8. Implementation of signal processing algorithms in mobile devices
9. Implementation of image processing algorithms in mobile devices
10. Implementation of signal and image processing in robotics

List of Home Assignments:

Design:

1. Design of FIR filter for given application
2. Design of IIR filter for given application
3. Design of image filtering algorithm
4. Design of speech algorithm
5. Design of image recognition algorithm

Case Study:

1. Signal Processing case study for audio signal processing
2. Signal Processing case study for biomedical signal processing
3. Signal processing case study in machine learning
4. Image processing case study in deep learning
5. Image processing case study in robotics

Blog

1. Application of signal processing in machine learning
2. Application of signal processing in medical equipment
3. Application of signal processing in robotics
4. Applocation of image processing in robotics
5. Application of signal and image processing in industrial automation

Surveys

1. Survey of algorithm for speech recognition
2. Survey of algorithms for filtering
3. Survey of algorithm for image recognition
4. Survey of algorithms for object detection
5. Survey of algorithms for image classification

Assessment Scheme:

- MSE
- ESE
- LAB assessment
- Course project assessment
- Presentation assessment
- Group discussion assessment
- Home assignment assessment
- Viva voce

Text Books:

1) J. G. Proakis & D. G. Manolakis, "Digital Signal Processing –Principles, Algorithms and Applications", Prentice Hall of India
2) Gonzalez Rafael C and Woods Richard E, Digital Image Processing, 3rd Edition, Prentice Hall, 2008.

Reference Books:

1. E. C. Ifeachor & B. W. Jarvis ,"Digital Signal Processing- A Practical Approach", Pearson Education.
2. Jain Anil K, Fundamentals of Digital Image Processing, PrenticeHall, 1989

Moocs Links and additional reading material: www.nptelvideos.in

1) Signal Processing - https://nptel.ac.in/courses/117/102/117102060/
2) Image Processing - https://nptel.ac.in/courses/117/105/117105135/

Course Outcomes: The student will be able to

IC2206_CO1: Digitize the continuous time signal, perform various operations on the signal and analyze the properties of the given digital systems. [3] (PO1, 2, 3, 4, 5. PSO-2,3).

IC2206_CO2: Recognize signal spectrum using DFT , compute and plot the spectrum using FFT , Analyze the signal in the frequency domain . [5] (PO1, 2, 3, 4, 5. PSO- 2, 3). Reconstruct time domain signal using IDFT Compute and plot time domain signal using IFFT Algorithm (PO1, 2, 3, 4, 5. PSO-2,3)

IC2206_CO3: Design FIR filters on paper to meet specific magnitude and phase requirements. Use MATLAB / Python / Scilab/ Octave to design the filters and analyze its response. Analyze the effect of quantization on the response of the digital filters [4] (PO1,2,3,4,5, PSO2, 3)

IC2206_CO4: Design IIR filters on paper to meet specific magnitude and phase requirements. Use MATLAB / Python / Scilab/ Octave to design the filters and analyze its response. Analyze the effect of quantization on the response of the digital filters [4] (PO1,2,3,4,5, PSO2, 3)

IC2206_CO5 - : Analyze images in the frequency domain using various transforms. [4] (PO1,2,3,4,5, PSO2, 3)

IC2206_CO6 - Evaluate the techniques for image enhancement and image restoration. Interpret image segmentation and representation techniques [4] (PO1,2,3,4,5, PSO2, 3)

CO PO Map

CO/ PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	3	3	3	3	3	-	-	-	-	-	-	2	-	3	3
CO2	3	3	3	3	3	-	-	-	-	-	-	2	-	3	3
CO3	3	3	3	3	3	-	-	-	-	-	-	2	-	3	3
CO4	3	3	3	3	3	-	-	-	-	-	-	2	-	3	3
CO5	3	3	3	3	3							2	1	1	2
CO6	3	3	3	3	3							2	-	1	2

CO attainment levels

CO NO	CO1	CO2	CO3	CO4	CO5	CO6
Attainment level	**3**	**5**	**4**	**4**	**4**	**4**

Future Courses Mapping:
Computer vision, machine learning, deep learning, artificial intelligence, natural language processing

Job Mapping:

Signal and Image processing course may lead to jobs in embedded system, digital system manufacturing industries, software industries for application development, finance software

FF No. : 654

IC2208 :: ENGINEERING DESIGN- II

Course Prerequisites:

No Prerequisites

Course Objectives: The student will be able to

1. Understand the importance of choosing socially relevant areas for project work
2. Understand the importance of Project centric learning
3. Plan and execute systematic strategy to complete the Project work
4. Document and present the completed project work in proper scientific format

Credits: 1

Teaching Scheme Theory: Hours/Week

Tut: Hours/Week

Lab: 2 Hours/Week

Course Relevance: This course will develop

1. Awareness about project centric learning will be quite useful in professional work in future
2. Self learning ability to up skill and upgrade once knowledge continuously
3. Ability to work in a Team and Team leadership which will be useful while doing B.Tech Major projects

Topics and Contents
Basics for Projects Importance of Project Centric Learning, Concept of Domains, Tools and Technology**,** Socially Relevant Project Areas **Domain Project Areas:** Awareness and identification of appropriate areas for project work such as: Agriculture, Defense, Healthcare, Smart city, Smart energy, Security Systems, Automobile, Space, Green Earth, Automobiles, Assistive Aid, Water Management, Swachh Bharat (any other socially relevant research area) ***Tools:*** *Self learning Activity* Learn and use latest engineering tools as per the project need. A few are listed below **Tools in Computer Engineering:** **Programming / Coding Tools** :- JavaScript, Python, Java, C#, C++, PHP, **Computer Vision Tools** :- OPENCV,MATLAB), **Single board computers:** Raspberry Pi, **Neural network simulators Tools:- Neural Lab, NEST , Machine Learning Tools:- Torch, TensorFlow, Data Science Tools :-** R language programming**,** SQL, **Tools in Electronics and Electronics & Telecommunication Engineering:** **Electronic Design Simulation Integrated Circuit Tools**:- VHDL, Xilinx, Modelsim , Cadence learn, **Embedded System Tools**:- AVR Studio, Arduino ,Kiel µvision, **Circuit Simulation Tools:-** Pspice, Simulink, Workbench, Tinkercad, ThingSpeak, Proteus, CircuitPro ,**Processor based integrated circuits** :Microcontroller, electronic prototype platforms: Arduino,**Networking Tools** :- Wired / Wireless and Ad-hoc Networking NS-2 , Packet Tracer, **Signal Processing Tools**:- Code Composer Studio along with Integrated circuits

Tools in Instrumentation and Control Engineering:-
System Automation Tools :- PLC , SCADA , PADS, ORCAD ,Eagle, Kicad,
Tools in Mechanical, Industrial, Production, Engineering:-
Engineering Design Tools:- AutoCAD, CATIA,COMSOL Multiphysics, Solidworks, Inventor, PTC Creo **Fluid Dynamics:-** Fluent, HyperWorks, **Finite Element/ Structural Analysis:-** Ansy's, Ansy's Free Student software **Thermal Simulation:-** FlowTherm, Ansys Icepak
Tools in Chemical Engineering :-
Chemical process simulator:- DWSIM - Open Source Process Simulator, **chemical simulation software:-** Schrödinger,
(any other suitable tool as per the project requirement)

Technology: Map the appropriate technology**:**
Emerging Technologies :- Artifical Intelligence, 5G networks, IoT, Serverless Computing, Blockchain , Virtual reality (VR)/Augmented reality (AR), Drone, Quantum Computing, Robotics

Interdisciplinary Technologies:- Nanotechnology, Nanomaterials, Nanoelectronics, Quantum Computing , Spintronics

Computer Technologies:- Big Data, Cloud Computing, Human Machine Interface (HMI),Cyber Security

Medical and Healthcare Technologies:- Biomedical Technology,

Energy Technologies :- Solar Energy Based Technologies, Wind energy, Green energy Technologies, Energy Storage
Electronics, Communication Technologies:- Wireless, GPS, Bluetooth, Mobile/social Internet Automation, Mobile Technologies, Voice Assistants, signal processing, image processing, Machine vision, Sensors, Optoelectronics,
Other imp Technologies:- Automobile ,3 D printing
(any other technology as per the project requirement)

Project Implementation: Selection of the domain area, Literature review, Identify and finalize the Problem Statement (student in consultation with Guide), Understand and select and use the appropriate tools, Map the technologies learned with the project needs (refer available online offline Resources, books, soft materials, relevant MOOCs, consult with domain expertise) Self Learning:- learn the required tools, skill sets, acquire knowledge to do the project

Designing & Testing: Designing of project prototype based on domain areas by incorporating appropriate tools and technology, validation and Testing of the prototype to give the best possible solution

Documentation and Final Assessment : Develop and demonstrate the optimized prototype /working model of project , Documentation of project report in stipulated standard format as per the preset norms i.e. IEEE Research paper format, Present Project work at final viva voce

FF No. : 654

IC2210:: SOFTWARE DESIGN – II

Guidelines to the students regarding Software Development Lab

The objectives of these courses are to enhance coding skills and programming ability among the students. To cope up with rapid technology changes, these courses offer every student to learn new-age programming techniques and languages. The focus of these courses is on learning programming fundamentals and techniques.

- Software Development course would be conducted as single student activity.
- Students can choose any software projects to upgrade and enhance their coding skills using any open source tools.
- Complexity of the project should be sufficient and approved by course supervisor.
- Students are allowed to use libraries as needed.
- Major thrust areas of Software Development Project course are Artificial Intelligence / Machine Learning / Data Analytics / Vision based Automation
- A suggestive list of possible domains for SDP is given below
    1. Mobile app development
    2. Responsive Web development
    3. Database / Back end development
    4. MySQL / RDBMS
    5. Gamification
    6. GUI Development
- Mid-Semester review and End Semester Assessment would be conducted.

Course Outcome: Students will be able to

1. Map the technologies learned with the project needs
2. Apply the technological knowledge to design various feasible solution
3. Select best possible solution to solve a the problem

FF No. : 654

IC2212:: SOFTWARE DESIGN PROJECT– II

Guidelines to the students regarding Software Development Project Course

The objectives of these courses are to enhance coding skills and programming ability among the students. To cope up with rapid technology changes, these courses offer every student to learn new-age programming techniques and languages. The focus of these courses is on learning programming fundamentals and techniques.

- Software Development course would be conducted as single student activity.
- Students can choose any software projects to upgrade and enhance their coding skills using any open source tools.
- Complexity of the project should be sufficient and approved by course supervisor.
- Students are allowed to use libraries as needed.
- Major thrust areas of Software Development Project course are Artificial Intelligence / Machine Learning / Data Analytics / Vision based Automation
- A suggestive list of possible domains for SDP is given below
    1. Mobile app development
    2. Responsive Web development
    3. Database / Back end development
    4. MySQL / RDBMS
    5. Gamification
    6. GUI Development
- Mid-Semester review and End Semester Assessment would be conducted.

Course Outcome: Students will be able to

1. Map the technologies learned with the project needs
2. Apply the technological knowledge to design various feasible solution
3. Select best possible solution to solve a the problem

FF No. : 654

IC2214:: ENGINEERING DESIGN AND INNOVATION-IV

Course Prerequisites: Electronic design, simulation, MATLAB, Labview, PCB design

Course Objectives:

1. To gain practical knowledge
2. To understand the concepts
3. Familiarity with the usage of modern tool

Credits: 4

Teaching Scheme Theory: - Hours/Week
Tut : - Hours/Week
Lab : 8 Hours/Week

Course Relevance:.....

SECTION-1
Topics and Contents **It is based on Real time project implementation in the chosen specific defined area.** Agriculture Healthcare Automotive Process Control IoT