DEPARTMENT OF TECHNOLOGY
SECOND YEAR B.TECH. (Computer Science and Technology)
Scheme of Teaching and Examination
Semester – III (Computer Science and Technology)

To be implemented from Academic Year 2021- 22

Sr. No.	Subject	Teaching Scheme (Hours / Week)				Examination Scheme (Marks)					
						Theory			Practical		
		L	T	P	Credits	Scheme	Max. marks	Min. Passing	Scheme	Max. marks	Min. Passing
MA211	Applied Mathematics-I	3	1	-	04	CIE	30	40	IOE	50	20
						SEE	70		-	-	-
CS211	Discrete Mathematical Structure	3	1	-	04	CIE	30	40	-	-	-
						SEE	70		-	-	-
CS212	Digital System and Microprocessor	4	-	-	04	CIE	30	40	-	-	-
						SEE	70		-	-	-
CS213	Data Structures	4	1	-	05	CIE	30	40	-	-	-
						SEE	70		-	-	-
CS214	Data Communication and Networking	4	-	-	04	CIE	30	40	-	-	-
						SEE	70		-	-	-
CS212L	Digital System and Microprocessor Lab	-	-	2	01	-	-	-	IPE	50	20
									EPE	50	20
CS213L	Data Structures Lab	-	-	4	02	-	-	-	IPE	50	20
									EPE	50	20
CS214L	Data Communication and Networking Lab	-	-	2	01				EPE	50	20
	Total	**18**	**03**	**08**	**25**	-	**500**	-	-	**300**	-
HS211	Environmental Studies	02	-	-	-	Project*	30	40	-	-	-
						Theory*	70				
Audit Course											
HS212	Introduction to Performing Arts	02	-	-	-	Evaluation at institute/ department level			Based on total marks obtained out of 50,the grade to be given by the course auditor		

Total Credits: 25 Total Contact Hours/Week: 33hrs

Note:

$ In theory student should appear for the CIE (Mid Semester Exam), submit the assignment and must secure 40% marks in SEE.

***** indicates Environmental Studies project evaluation and the theory examination will be at the end of the year i.e. along with Semester IV End Examination.

CIE– Continuous Internal Evaluation (Mid Semester Evaluation),

SEE – Semester End Examination,

IPE – Internal Practical Evaluation, EPE–External Practical Examination,

IOE– Internal Oral Evaluation, EOE–External Oral Examination

DEPARTMENT OF TECHNOLOGY
SECOND YEAR B.TECH. (Computer Science and Technology)
Scheme of Teaching and Examination
Semester – IV (Computer Science and Technology)
To be implemented from AcademicYear 2021-22

Sr. No	Subject	Teaching Scheme (Hours / Week)				Examination Scheme (Marks)					
						Theory			Practical		
		L	T	P	Credits	Scheme	Max. marks	Min. Passing	Scheme	Max. marks	Min. Passing
CS221	Theory of Computation	3	1	-	04	CIE	30	40	-	-	-
						SEE	70		-	-	-
CS222	Advanced Microprocessor	3	1	-	04	CIE	30	40	-	-	-
						SEE	70		-	-	-
CS223	Computer Organization	3	-	-	03	CIE	30	40	-	-	-
						SEE	70		-	-	-
CS224	Software Engineering	3	-	-	03	CIE	30	40	-	-	-
						SEE	70		-	-	-
CS225	Applied Mathematics-II	3	1	-	04	CIE	30	40	-	-	-
						SEE	70		-	-	-
CS222L	Advance Microprocessor Lab	-	-	2	01	-	-	-	IPE	50	20
									EPE	50	20
CS226L	Linux and Shell Programming	2	-	2	03	-	-	-	IPE	50	20
									EPE	50	20
CS227L	Object Oriented Programming Lab	2	-	2	03	-	-	-	IPE	50	20
									EPE	50	20
	Total	**19**	**03**	**06**	**25**	-	**500**	-	-	**300**	-
HS211	Environmental Studies	02	-	-	-	Project	30	40	-----	----	---
						Theory	70				
Audit Course											
HS222	Soft Skills Development	02	-	-	-	Evaluation at institute/ department level		Based on total marks obtained out of 50,the grade to be given by the course auditor			

Total Credits: 25 Total Contact Hours/Week: 32hrs

Note:

$ In theory student should appear for the CIE (Mid Semester Exam), submit the assignment and must secure 40% marks in SEE.

CIE– Continuous Internal Evaluation (Mid Semester Evaluation)
SEE – Semester End Examination,
IPE – Internal Practical Evaluation, EPE–External Practical Examination,
IOE– Internal Oral Evaluation, EOE–External Oral Examination

Internship I which is a part of Semester V evaluation will be the activity after the SEE of semester IV. It is mandatory for all the students to undergo Internship I and report to the institute for the semester V along with the completion certificate by the concerned organization. The students have to submit a hard as well as soft copy of the activity report to the institute.

Class, Part & Semester	**:**	**Second Year B. Tech (Computer Science and Technology), Part II, Sem- III**					
Course Title	**:**	**Applied Mathematics – I**			***Course Code*:**	**:**	**MA211**
Teaching Scheme (Hours)	**:**	Lecture : 3 Hrs/week Tutorial : 1 Hrs/week			***Total Credits***	**:**	04
Evaluation Scheme (Marks)	**:**	CIE=30 (20+10) IOE=50	SEE = 70	Grand Total=100	***Duration of SEE***	**:**	3 hrs
Revision:	**:**	Fourth			***Month***	**:**	June 2021
Pre-requisites *(if any)*	**:**	Basic knowledge of Engineering Mathematics-I and Engineering Mathematics-II.					
Course Domain	**:**	Basic Sciences					

Course Rationale:

This course offers a mathematical understanding for engineering applications. This course produce graduates with mathematical knowledge, computational skills and the ability to deploy these skills effectively in the solution of problems, principally in the area of engineering.

Course Objectives: The Course teacher will		***Course Outcomes:*** Students will be able to	
1.	Introduce linear differential equations and partial differential equations.	**1.**	Solve linear differential equations and apply them on simple electric circuit.
2.	Explain Laplace Transform, Inverse Laplace Transform and applications to electric circuit problems	**2.**	Solve the problems on partial differential equations.
3.	Demonstrate Fourier transform and their applications.	**3.**	Gain the basic knowledge of Laplace transform and their applicability in solving initial value problems.
4.	Explain mathematical programming and assignment problems.	**4.**	Understands the new notion of Fourier transform and their usability.
5.	Demonstrate applications to computer engineering.	**5.**	To solve engineering problems using Mathematical Programming.
		6.	Analyze and solve engineering problems using Assignment problems.

Curriculum Content	**Hours**
Unit I Linear Differential Equations Linear Differential Equations with constant coefficients, Homogeneous Linear differential equations, Applications of LDE with constant coefficients to Electrical systems.	7
Unit II Partial Differential Equations First order partial differential equations, solutions of first order linear and non-linear PDEs- Four	6

standard forms of partial differential equations of first order.	
Unit III Laplace Transform: Definitions, Laplace transform of standard functions, Properties & theorems of Laplace transform, Inverse Laplace transform and application to solutions of linear differential equations (electric circuit problems).	7
Unit IV Fourier Transform: Definition, Properties & theorem, Fourier sine & cosine transform, Inverse Fourier transform, Discrete Fourier transform & its properties, Applications of Fourier transform.	7
Unit V Mathematical Programming: Linear Optimization problems, Standard and Canonical forms, Basic solutions and feasible solutions, Optimal solutions by simplex method**,** Big M-method, Relation between Primal and Dual L.P.P., Dual simplex method, Solution of Primal L. P. P. using Dual L. P. P.	6
Unit VI Assignment Problems: Definition, Balanced and Unbalanced assignment problems, Hungarian method of solving assignment problems. Travelling salesmen problem.	6

Suggested list of Tutorials and Assignments:

1. To find solution of LDE with constant coefficients
2. Examples of Homogeneous LDE
3. Problems on Partial differential equations
4. Examples on Properties of Laplace transform
5. Examples on Inverse Laplace transform
6. Examples on Fourier transform
7. Examples on Simplex and Dual Simplex method
8. Examples on Big M-method
9. Assignment Problems

General Instructions:

1. Batch wise tutorials are to be conducted. The number of students per batch should be as per the practical batches.
2. Students must be encouraged to solve engineering mathematics problems using different software's in tutorial class only.
3. Each Student has to write at least 6 assignments on entire syllabus.

Suggested Text Books:

1.	Erwin Kreyszig, "Advanced Engineering Mathematics", Fifth Edition, John Wiley & Sons.
2.	B. S. Grewal, "Higher Engineering Mathematics", 43rd Edition, Khanna Publishers, Delhi.
3.	S. D. Sharma, "Operations Research", 11th Edition.

Suggested Reference Books:

1.	C. R. Wylie, "Advanced Engineering Mathematics", 6th Edition , McGraw Hill Publication, New Delhi.
2.	H. A. Taha, "Operations Research", 8th Edition, Pearson.

3.	S. S. Sastry, "Engineering Mathematics (Volume-I)", 4th Edition, Prentice Hall Publication, New Delhi.
4.	H. K. Dass, "Advanced Engineering Mathematics", 2014, S. Chand Publishing.
5.	N. P. Bali, Iyengar "A text book of Engineering Mathematics by", Laxmi Publications (P)Ltd., New Delhi.
6.	M. D. Greenberg, "Advanced Engineering Mathematics", 2nd Edition, Pearson Education.

Class, Part & Semester	**:**	**Second Year B. Tech (Computer Science and Technology), Part II, Sem-III**					
Course Title	**:**	**Discrete Mathematical Structure**			***Course Code*:**	**:**	CS 211
Teaching Scheme (Hours)	**:**	Lecture : 3 Hrs/week Tutorial : 1 Hrs/week			***Total Credits***	**:**	4
Evaluation Scheme (Marks)	**:**	CIE=30 (20+10)	SEE = 70	Grand Total=100	***Duration of SEE***	**:**	3 hrs
Revision:	**:**	Fourth			***Month***	**:**	June 2021
Pre-requisites *(if any)*	**:**	Basic Mathematics					
Course Domain	**:**	Core (**(Mathematical Logic, Set theory, Algebraic Structures, Boolean algebra, Graph Theory)**					

Course Rationale:

Discrete mathematics forms the mathematical foundation of computer science and technology. Learners will become familiar with a broad range of mathematical objects like sets, functions, relations, graphs, that are ever-present in computer science. Concepts and notations from discrete mathematics are useful in studying and describing objects and problems in all branches of computer science, such as computer algorithms, programming languages, cryptography, automated theorem proving, and software development.

Course Objectives: The Course teacher will		***Course Outcomes:*** Students will be able to	
1.	Introduce most of the basic terminologies used in computer science courses and application of ideas to solve practical problems.	**1.**	Apply mathematical thinking, mathematical proofs, and algorithmic thinking, and be able to apply them in problem solving
2.	Explain basic mathematical logic and Set theory.	**2.**	Demonstrate the fundamental concepts related to set theory, relations and functions which are frequently required in advanced courses such as analysis of algorithms.
3.	Demonstrate relations and functions	**3.**	Compare algebraic structures like monoid, semigroups and groups.
4.	Extend student's Logical and Mathematical ability to deal with abstraction	**4.**	Learn and summarize the group theory and group codes with applications in communication model.

5.	Expose to concepts and properties of algebraic structures such as semi groups, monoids and groups	**5.**	Develop the ability to solve the problems related to algebra, POSETs, lattices, Boolean algebra and their application in computer science.
6.	Demonstrate core ideas in graph theory	**6.**	Solve the practical problems using graphs and related discrete structures

Curriculum Content	**Hours**
Unit I Mathematical Logic Introduction, statements and Notation, Connectives, statement formulas and truth tables, well-formed formulas, Tautologies, Equivalence of formulas, Duality law, Tautological implications, functionally complete sets of connectives, other connectives, Normal & Principle normal forms.	8
Unit II Set Theory Basic concepts of set theory, types of operations on sets, ordered pairs, Cartesian product, representation of discrete structures, relation, properties of binary relations, matrix and graph representation, partition and covering of set, equivalence relation, composition, POSET and Hasse diagram, Function – types, composition of functions, Inverse function.	9
Unit III Algebraic Systems Semigroups and Monoids, properties and examples.	3
Unit IV Groups Definition and examples, subgroups and homomorphism, Group codes, communication model, Generation of codes using checksum, error recovery in group codes.	4
Unit V Lattices and Boolean Algebra Lattice as POSETs, definition, examples and properties, Lattice as algebraic systems, Special lattices, Boolean algebra definition and examples, Boolean functions, representation and minimization of Boolean functions.	7
Unit VI Graph Theory Basic concepts of graph theory, Storage representation and manipulation of graphs, Fault detection in combinational switching circuits – Faults in combinational circuits, Notions of Fault detection, Algorithm for fault matrix, PERT and related techniques.	8

Suggested list of Tutorials and Assignments:

1. Equivalence of formulas
2. Normal & Principle normal forms
3. Relations and properties
4. POSET and Hasse diagram
5. Functions
6. Properties of Semigroups and Monoids
7. Group and Group codes.
8. Lattice and properties
9. Boolean algebra and properties
10. Storage representation and manipulation of graphs
11. PERT

Suggested Text Books:	
1.	Discrete mathematical structures with applications to computer science", J. P. Tremblay& R. Manohar, Tata McGraw-Hill Edition, 35th Reprint
2.	"Elements of Discrete Mathematics", C. L. LIU, Tata McGraw-Hill, 2nd Edition, 2002, ISBN 0- 07-043476-X.
Suggested Reference Books:	
1.	"Discrete Mathematics and Its Applications", Kenneth H. Rosen, Tata McGraw-Hill, 5th Edition, 2003, ISBN 0-07-053047-5.
2.	"Theory and problems in Abstract algebra",Schaums outline series, MGH.
3.	"Discrete Mathematics",Lipschutz, Lipson, Tata McGraw-Hill, 2nd Edition, 1999, ISBN 0-07-463710--X.
4.	"Graph Theory", V. K. Balakrishnan, TMH (Recommended for Graph) ISBN 0-07-058718-3
5.	"Discrete Mathematical Structures", B. Kolman, R. Busby and S. Ross, Pearson Education, 4th Edition, 2002, ISBN 81-7808-556-9

Class, Part & Semester	**:**	**Second Year B. Tech (Computer Science and Technology) Part II, Sem III**					
Course Title	**:**	**Digital Systems and Microprocessor**			***Course Code*:**	**:**	**CS 212**
Teaching Scheme (Hours)	**:**	Lecture:	4 Hrs/week		***Total Credits***	**:**	4
		Tutorial:	----				
Evaluation Scheme (Marks)	**:**	CIE=30 (20+10)	SEE = 70	Grand Total=100	***Duration of SEE***	**:**	3 hrs
Revision:	**:**	Fourth			***Month***	**:**	*February 2021*
Pre-requisites *(if any)*	**:**	Basic knowledge of digital logic and computer hardware basics.					
Course Domain	**:**	**Core (Logic gates, Boolean algebra, Microprocessors)**					

Course Rationale: A study of digital systems and the building blocks that make up digital systems. The emphasis will be on microprocessor-based systems hardware, programming and interfacing

Course Objectives: The Course teacher will		***Course Outcomes:*** Students will be able to	
1.	Introduce the analysis and design of digital systems and microprocessors.	**1.**	Understand the logical behaviour of digital circuits
2.	Review combinatorial analysis and design.	**2.**	Design combinational logic using Karnaugh maps
3.	Explain Computer aided design and programming of digital electronic circuits through the application of several modern software packages.	**3.**	Design sequential logic using ASM charts
4.	Demonstrate analysis and design of synchronous finite state machines and register transfer level systems.	**4.**	Analyse combinational and sequential digital circuits
5.	Introduce microprocessor devices, their architecture and instruction sets, Hardware aspects of instruction execution, Assembly language programming.	**5.**	Explain the architecture, pin configuration of various microprocessors
6.	Explain input/output, bus interfacing, interrupts and co-design of digital hardware and microprocessor systems.	**6.**	Perform various microprocessor-based programs and apply the concepts of 8085 programming, interrupts, stacks & subroutines

Curriculum Content	**Hours**
Unit: I Fundamentals Concepts: Logic Families, TTL, TTL sub families, Characteristics of TTL gates, Axioms and laws of Boolean algebra, Practical examples with logic gates IC's.	3
Unit: II Combinational Logic Design: Boolean algebra, min and max terms, K-maps and quine –McClusky methods, Solution using K-maps, SOP & POS representation of digital logic and their reduction using K-map, BCD to 7-segment converter, Multiplexer and demultiplexer, encoder, decoder, Half and Full adder design	8

using gates.	
Unit: III Sequential Logic Design: Various flip flops (R-S, D, J-K, T) using gates, counter using J-K flip-flops, shift Register using flip-flops, study of different ICs (7490, 7495, 74LS138, 7447) Timer IC (555), IEEE / ANSI symbols **Analog Electronics:** OP-AMP (741), Basics of OP-AMP, Characteristics, Adder, Substractor, Integrator, Differentiator, Comparator using OP-amp	8
Unit: IV 8085 Microprocessor Introduction: Introduction to Microprocessor, Features of 8085, 8085-CPU architecture, Demultiplexing of address and data bus, Instruction fetching and execution operation of microprocessor.	4
Unit: V 8085 Instruction Set: Instruction formats, addressing modes, Op-code formats, Classification of Instruction set, Programming technique, Instruction timings, WAIT state, Single step and single cycle execution.	8
Unit: VI Interrupt and DMA Transfer: Types of Memory, Memory organizations Mapping of I/O 8085 Interrupts RST5.5, RST6.5, RST7.5, TRAP & INTR. Designing hardware for INTR, Interrupt priorities, SIM and RIM instruction, DMA transfer, HOLD and HLDA pins for DMA transfer. **I/O Operation and interfacing:** Devices, IN & OUT Instruction with timing diagrams study of 8255 PPI, Interfacing Keyboards, Interfacing Thumbwheel switches, 8253.	8

Suggested list of Tutorials and Assignments:

General Instructions: ---Student Should Complete one assignments per Unit

Suggested Text Books:	
1.	*"Modern Digital Electronics" 4th Edition,ByR.P.Jain*
2.	*"Microprocessor Architecture Programming & Application",* Ramesh Gaonkar, Willey Estern.5th Edition
3.	*"Digital Systems-Principals and Application*", Tocci, Widmer, Moss, (Pearson Education) 11th Edition
4.	*"Design with operational amplifier",* Sergio Franko and book by RamakantGaiekwad4th Edition
Suggested Reference Books:	
1.	*"Fundamentals of digital circuits",* B.Anandkumar 4th Edition
2.	*"Digital Systems & Microprocessor",* Douglas Hall MGH3rd Edition
3.	"Digital Logic and Computer Design",Book by M. Morris Mano 5th Edition

Class, Part & Semester	**:**	**Second Year B. Tech (Computer Science & Technology), Part II, Semester III**					
Course Title	**:**	**Data Structures**			***Course Code*:**	**:**	**CS 213**
Teaching Scheme (Hours)	**:**	Lecture :	4Hrs/week		***Total Credits***	**:**	05
		Tutorial :	1 Hrs/week				
Evaluation Scheme (Marks)	**:**	CIE=30 (20+10)	SEE = 70	Grand Total=100	***Duration of SEE***	**:**	3 Hrs
Revision:	**:**	Fourth			***Month***	**:**	June 2021
Pre-requisites *(if any)*	**:**	Basic understanding of C programming language and basic mathematics.					
Course Domain	**:**	**Core (Data Structure)**					

Course Rationale: Study of data structure is an essential part of computer science. In application programming and system programming data structures methods & techniques are widely used. The data structure study helps the students for developing logic & structured programs.

Course Objectives: The Course teacher will		***Course Outcomes:*** Students will be able to	
1.	Introduce the fundamental concept and importance of data structures in developing and implementing efficient algorithms.	**1.**	To analyze the concepts of data structure and data type.
2.	Demonstrate the implementation of linked data structures such as arrays, stacks and queues.	**2.**	Develop knowledge of basic data structures for storage and retrieval of ordered or unordered data.
3.	Demonstrate the data structures such as linked lists, hash tables.	**3.**	Implement linked list data structure to solve various problems.
4.	Explain searching and sorting techniques operations with examples.	**4.**	Understand and apply various data structure such as stacks, queues, trees and graphs to solve various computing problems using C-language.
5.	Introduce the concepts of trees and graph with operation.	**5.**	Develop knowledge of applications of data structures including the ability to implement algorithms for the creation, insertion, deletion, searching, and sorting of each data structure.
6.	Explain the basic concept of graph theory	**6.**	Understand the concepts of graph theory.

Curriculum Content	**Hours**

UnitI : Stacks and Queue Fundamentals stack and queue as ADT, Representation and Implementation of stack and queue using sequential and linked organization, circular queue: representation and implementation, Application of stack for expression evaluation and for expression conversion, Recursion, Priority queue, Doubly Ended Queue.	7
Unit II : Searching and sorting **Search:** Importance of searching, Sequential, Binary, Fibonacci search algorithms **Sorting:** Quick sort, two-way merge sort, heap sort, shell sort, Radix sort.	9
Unit III : Linked list Concept of linked organization, Singly linked list, doubly linked list and dynamic storage management, circular linked list, Operations such as Insertion, deletion, inversion, concatenation, Computation of length, traversal on linked list, Representation & manipulations of polynomials using linked lists.	7
Unit IV : Hashing Definition, Hash functions, Overflow, Collision, Open Hashing, closed hashing, Rehashing Techniques.	6
Unit V : Tree Basic Technology, Binary Tree, Traversal methods, Binary search tree, B tree, B+ tree, Heaps - operations and their applications.	8
Unit VI : Graph Basic concepts of graph theory, storage representation and manipulation of graphs, Introduction to Sparse matrix, representation of sparse matrix using linked list.	7

Suggested list of Tutorials and Assignments:

1. Explain with example and with operations stack, Queue
2. Explain with example quick sort, merge sort, Radix sort, shell sort, heap sort
3. Explain with example and with operations Single linked list, doubly linked list
4. Explain with example open hashing, closed hashing, Rehashing techniques
5. Explain with example Binary tree, BTree, B+ tree, Tree traversal
6. Explain sparse matrix using linked list,Graph traversing techniques

General Instructions: Student is evaluated during Continuous Internal Evaluation (Internal Test), Internal practical Examination and Semester End Examination.

Suggested Text Books:

1.	Data Structure using C -- A. M. Tanenbaum, Y. Langsam, M. J. Augenstein (PHI). 2nd Edition
2.	Data Structures using C – ISRD Group, TMH publication 2nd Edition

Suggested Reference Books:

1.	Data structures and Algorithms -- Alfred V. Aho, John E. Hopcroft, J. D. Ullman (Addision-Wesely Series)
2.	Data structures -- Seymour Lipschutz (MGH) Schaum's Outlines.4th Edition
3.	Introduction to Data Structures in C – Ashok N. Kamthane (Pearson Education). 2nd Edition
4.	Data Structures- A Pseudo code Approach with C – Richard F. Gilberg and Behrouz A. Forouzon 2nd Edition

Class, Part & Semester	**:**	**Second Year B. Tech (Computer Science and Technology), Part II, Sem- III**					
Course Title	**:**	**Data Communication and Networking**			***Course Code*:**	**:**	**CS214**
Teaching Scheme (Hours)	**:**	Lecture :	4 Hrs/week		***Total Credits***	**:**	4
		Tutorial :	--				
Evaluation Scheme (Marks)	**:**	CIE=30 (20+10)	SEE = 70	Grand Total=100	***Duration of SEE***	**:**	3 hrs
Revision:	**:**	Fourth			***Month***	**:**	June 2021
Pre-requisites *(if any)*	**:**	---					
Course Domain	**:**	**Core (Networking)**					

Course Rationale: To develop an understanding of the various aspects of data communications and computer networking systems.The subject will introduce the basics of data communications and computer networks. In this students will examine and understand network protocols and architectures. It will educate the student with modern networking technologies.

Course Objectives:The Course teacher will		***Course Outcomes:*** Students will be able to	
1.	Provide knowledge about basics of Data Communication and computer network	**1.**	Explain Data Communications System, its components and Articulate the networking Basics.
2.	Provide knowledge about the Functions of Physical Layer.	**2.**	Differentiate and analyze various multiplexing techniques. Explain and examine wired and wireless communication with medium access control layer.
3.	Provide details of different data link layer functions including error detection and error corrections.	**3.**	Differentiate various encoding techniques. Apply error control techniques.
4.	Provide knowledge about different framing techniques and network layer protocols for data communication.	**4.**	Solve sub-netting problems and analyze various routing mechanism, Identify and compare congestion control mechanisms
5.	Provide detail knowledge of Transport Layer and protocols.	**5.**	Examine the services provided by transport layer and have a hands-on experience of socket programming
6.	Provide knowledge about protocols from application layer.	**6.**	Inspect the networking applications used in everyday tasks such as reading email or surfing the web and analyze its architecture

Curriculum Content	**Hours**

Unit 1: Network Architectures **Introduction :** Data communications, Networks, Protocols & standards, Introduction to computer networks, LAN, MAN, WAN, VAN **Network topologies**: Bus, Star, Ring, Mesh, Hybrid, Types of networks **Layered network model:** OSI, TCP/IP, ATM model	09
Unit II: Physical Layer Characterization Introduction to physical layer, **Transmission media:-** Guided and Unguided media, Networking Hardware components, Transmission modes, **Introduction to packet switching:** Circuit switching vs. packet switching, **Types of services: -** Connection oriented services, Connectionless services, Multiplexing.	08
Unit III: Data Link Layer **Error detection & correction:** cyclic codes, Hamming code, **Data Link Control: -** Farming, Flow & error control, stop & wait protocol, sliding window protocol, HDLC protocol.	09
Unit IV: Network Layer, Internet Protocol, Routing Protocols **IPv4 Addresses:** Introduction, Classful and Classless Addressing, Special Addresses, Network Layer Design Issues **Routing Algorithms:** Shortest Path, Flooding, Distance Vector, Link State Routing, **Congestion control:** Congestion prevention policies, congestion control in datagram subnet, congestion control in diagram subnet, Load Shedding, Jitter Control.	09
Unit V: Transport Layer The Transport service primitives, **UDP:** Process to Process communication, User Datagram Format, Operation and uses of UDP. **TCP:** TCP Services and Features, TCP segment format, TCP Connections, Flow and error control in TCP, TCP Timers. **Berkeley Sockets:** Socket Addresses, Elementary Socket system calls byte ordering and address conversion routines, connectionless iterative server, Connection Oriented concurrent server, TCP and UDP Client server Programs.	09
Unit VI: Application Layer Name space, Domain Name Space, Distribution of Name Space, DNS in the Internet, Resolution, DNS message , Remote Login (SSH), Electronic mail, FTP, WWW & HTTP	08

Suggested Text Books:	
1.	B. A. Forouzan, "Data Communications and Networking", 4th Edition, Tata McGraw-Hill, 2013, ISBN-10: 1-25-906475-1
2.	Computer Networks – Andrew S. Tanenbaum (Pearson Education) 4th Edition
Suggested Reference Books:	
1.	William Stallings, "Data and computer Communication", 7th Edition, Pearson Education, 2003, ISBN-13: 978-0131006812, ISBN-10: 0131006819.
2.	Larry L. Peterson and Bruce S. Davie, "Computer Networks a systems approach", 5th Edition, Morgan Kaufmann an imprint of Elsevier, 2014, ISBN: 978-93-80501-93-2

Class, Part & Semester	**:**	**Second Year B. Tech (Computer Science and Technology) Part II, Sem III**					
Course Title	**:**	**Digital Systems and Microprocessor Lab**			***Course Code*:**	**:**	**CS 212L**
Teaching Scheme (Hours)	**:**	Practical:	2 Hrs/week		***Total Credits***	**:**	1
Evaluation Scheme (Marks)	**:**	IPE= 50	EPE=50	Total=100	***Duration of EPE***	**:**	**3 hours**
Revision:	**:**	Fourth			***Month***	**:**	February 2021
Pre-requisites *(if any)*	**:**	The prerequisite for this course is basic knowledge of digital system and microprocessors theory					
Course Domain	**:**	**Core (Logic gates, Boolean algebra, Microprocessors)**					

Course Rationale: The student will learn to program an 8-bit microprocessor in assembly language, and will develop the hardware and software for microprocessor-controlled applications.

Course Objectives: The Course teacher will		***Course Outcomes:*** Students will be able to	
1.	Introduce to the analysis and design of digital systems and microprocessors.	**1.**	Understand the logical behavior of digital circuits
2.	Review of combinational analysis and design.	**2.**	Design combinational logic using Karnaugh maps
3.	Do designing and programming of digital electronic circuits through the application of several modern software packages.	**3.**	Analyse combinational and sequential digital circuits
4.	Analyze and design of synchronous finite state machines and register transfer level systems.	**4.**	Design combinational and sequential digital circuits
5	Demonstrates Microprocessor devices, their architecture and instruction sets, Hardware aspects of instruction execution, Assembly language programming.	**5.**	Explain the architecture, pin configuration of various microprocessors
6.	Teach Input/output, bus interfacing, interrupts. Co-design of digital hardware and microprocessor systems.	**6.**	Apply the concepts of 8085 programming, interrupts, stacks & subroutines

List of Experiments

Sr. No.	**Name of Experiment**
1.	Study of Basic gates.
2.	Study of Universal gates

3.	Study of Boolean algebra & De Morgan's theorem using gates.
4.	Study of MUX/DEMUX.
5.	Study of 74138.
6.	Study of R-S and J-K flip-flops.
7.	Study of counters
8.	Interfacing of counters to seven segment display.
9.	Realization of 4/5 variable K-maps
10.	Study of 8085.
11.	Assembly language programming for 8085 (Arithmetic, Logical and data transfer-Minimum 8 programs).
12.	Writing subroutine to perform delay operation of 10 ms.
13.	Designing & implementing hardware for INTR
14.	Study of 8255. Interfacing using 8255.
15.	Study of 8253 interfacing.
General Instructions: Students have to perform 8-10 practicals from the list	
Suggested Text Books/ Reference Books/Manual	
1.	"Digital Logic and Computer Design"Book by M. Morris Mano5th Edition
2.	"Fundamentals of logic design"Book by Charles H Roth7th edition
3.	Microprocessor Architecture ... — Microprocessor Architecture, Programming and Applications with the 8085 written by Ramesh .
4.	*"Fundamentals of digital circuits",* B.Anandkumar4th edition

Class, Part & Semester	**:**	**Second Year B. Tech (Computer Science and Technology) Part II, Sem III**					
Course Title	**:**	**Data Structures Lab**			***Course Code*:**	**:**	**CS 213L**
Teaching Scheme (Hours)	**:**	Practical :	4 Hrs/week		***Total Credits***	**:**	2
Evaluation Scheme (Marks)	**:**	IPE=50	EPE=50	Total=100	***Duration of EPE***	**:**	03 hours
Revision:	**:**	Fourth			***Month***	**:**	June 2021
Pre-requisites *(if any)*	**:**	Knowledge of Programming Methodology, 'C' language, Control Statements, Functions, Arrays, Pointers, Structures and Union and File Handling concepts.					
Course Domain	**:**	Core (Data Structures)					

Course Rationale: TheCourse is designed to enablethe students to gain the practical applications knowledge of data structures and to develop skills to design and analyze linear and nonlinear data structures. It strengthen the ability of students by identifying and applying suitable data structure for the given real world problem.

Course Objectives:The Course teacher will		***Course Outcomes:*** Students will be able to	
1.	Dwemonstrate to design and analyze simple linear and non linear data structures.	**1.**	Understand the importance of data structure and abstract data type, and their basic usability in different applications through different programming languages.
2.	Explain to understand how several fundamental algorithms work particularly those concerned with various Sorting algorithms.	**2.**	Analyze and differentiate different algorithms based on their time complexity.
3.	Identify and apply the suitable data structure for the given problem.	**3.**	Analyze and differentiate different algorithms based on their time complexity.
4.	Teach to design and evaluate ADTs, nonlinear temporary and persistent data structures and also related algorithms.	**4.**	Design new algorithms or modify existing ones for new applications and able to analyze the space & time efficiency of most algorithms.
5	Improve the logical ability.	**5.**	Have practical knowledge on the application of data structures.
6	Help to Gain knowledge in practical applications of data structures.	**6.**	Be familiar with various data structure such as stacks, queues, trees, graphs, etc. to solve various computing problems.

List of Experiments

Sr. No.	**Name of Experiment**

1.	Write a program for matrix Manipulation using array.
2.	Implement Tower of Hanoi problem using recursion.
3.	Implement different operations on string without using library function.
4.	Implementation of palindrome string.
5.	Implement different operation on file.
6.	Implement stack as an ADT. Perform push() and pop() operations on it.
7.	Implementation of queue using array.
8.	Implement circular queue and double ended queue using arrays.
9.	Write a program for sequential search and linear search.
10.	Apply following searching techniques on list or array: Binary ii) Fibonacci
11.	Implement following sorting techniques on list or array: i) Quick sort ii) Merge sort.
12.	Write a program to create linked list and perform operation such as insert, delete, update, reverse.
13.	To implement of binary tree traversal.
14.	To study hashing techniques.
15.	To study graph traversal method.
General Instructions: Practical Journal Assessment, Internal practical Examination and ExternalPractical Examination	
Suggested Text Books/ Reference Books/Manual	
1.	Data Structure using C -- A. M. Tanenbaum, Y. Langsam, M. J. Augenstein (PHI).
2.	Data Structures- A Pseudo code Approach with C – Richard F. Gilberg and Behrouz A. Forouzon 2nd Edition
3.	Data structures -- Seymour Lipschutz (MGH) Schaum's Outlines.
4.	Data structures and Algorithms -- Alfred V. Aho, John E. Hopcroft, J. D. Ullman (Addision-Wesely Series)

Class, Part& Semester	**:**	**Second Year B. Tech (Computer Science and Technology), Part II, Sem - III**					
Course Title	**:**	**Data Communication and Networking Lab**			***Course Code*:**	**:**	**CS214L**
Teaching Scheme (Hours)	**:**	Practical :	2 Hrs/week		***Total Credits***	**:**	01
Evaluation Scheme (Marks)	**:**	IPE/IOE= Nil	EPE= 50	Total=50	***Duration of EPE***	**:**	03 hours
Revision:	**:**	Fourth			***Month***	**:**	June 2021
Pre-requisites *(if any)*	**:**	Knowledge of Programming Methodology, 'C' language, Control Statements, Functions, Arrays, Pointers, Structures and Union and File Handling concepts					
Course Domain	**:**	Core					

Course Rationale: In this lab demonstrate basics of data communication and networking concepts. Here concept of data transfer between nodes is introduced to students. In this working principle of various communication protocols is introduced. It includes analysis of the various routing algorithms.

Course Objectives:The Course teacher will		***Course Outcomes:*** Students will be able to	
1.	Explain theoretical and practical knowledge in computer networks.	**1.**	Demonstrate the practical aspect of networking related to the theoretical concepts.
2.	Demonstrate different framing techniques and error detection and correction code	**2.**	Demonstrate and Simulate Error Detection and correction code
3.	Help to Distinguish and show how to design and analyze different types of communication protocols.	**3.**	Implement basic protocols and socket programming
4.	Teach to interpret basic skills needed to write network application using socket interface	**4.**	Simulate, configure and analyze the network using networking tools

List of Experiments

Sr. No.	**Name of Experiment**
1.	Study and demo of LAN, WAN and various connecting devices and components.
2.	Study of Different Networking Command
3.	Implementation of Framing Method By Character Count

4.	Implementation of Error Detecting Code (CRC)
5.	Implementation of Error Correcting Code (Hamming Code).
6.	Implementation of Simplex Stop and Wait Protocol.
7.	File transfer using Go back n / Selective Repeat Protocol
8.	Implementation of Shortest Path algorithm
9.	Implementation of connection oriented (TCP) client-server socket program.
10.	Implementation of connectionless (UDP) client-server socket program.
11.	Study of network protocol analyzer (Wire-Shark) / (Packet sniffer) and understanding packet formats for UDP, TCP, ARP, ICMP protocols
12.	DNS client utilities with Nslookup and Dig
13.	Implement simple web page design
14.	Case study of campus-wide network
General Instructions: Students have to perform 8-10 practicals from the list	
Suggested Text Books/ Reference Books/Manual	
1.	Richard Steven, "Unix network programming", for Socket Programming, Prentice Hall 3rd edition, 2015
2.	James F. Kurose and Keith W. Ross, "Computer Networking: A Top-Down Approach Featuring the Internet", Pearson Education,5th /6th edition, 2012/2013
3.	Jeffery S. Beasley, "Networking", New Riders Press, 2nd edition, 2008.

Class, Part & Semester	**:**	**Second Year B. Tech (Computer Science and Technology), Part II & Sem-III**						
Course Title	**:**	**Environmental Studies**				***Course Code***	**:**	**HS211**
Teaching Scheme (Hours)	**:**	Lecture : 02 Hours/Week Tutorial : 00 Hours/Week				***Total Credits***	**:**	Nil
Evaluation Scheme (Marks)	**:**	CIE = 00 SEE = 70	IPE=30 Project	:	Grand Total=100	***Duration of SEE***	**:**	3 hrs. At the year end
Revision	**:**	Fourth				***Month***	**:**	June 2021
Pre-requisites *(if any)*	**:**	Completion of First Year Engineering, Revision of BS-12A2 namely Engineering Chemistry may help for better understanding.						
Course Domain	**:**	Ethics and Environment						

Course Rationale: The Course is all about learning the way we should live and how we can develop sustainable strategies to protect the environment. It helps individuals to develop an understanding of living and physical environment and how to resolve challenging environmental issues affecting nature.

Course Objectives: The Course Teacher will		***Course Outcomes:*** Students will be able to	
1.	Define the course and indicate the importance of the same to the students.	**1.**	Recognize the scope and need of the course.
2.	Enumerate the natural resources and make students visualize about associated problems.	**2.**	Identify the natural resources and detect the associated problems.
3.	Describe and relate the ecosystems the engineering graduates.	**3.**	Relate values of ecosystems to human, plants and animals.
4.	Explain concepts and theory in biodiversity and management from interdisciplinary perspectives.	**4.**	Identify key threats of biodiversity.

Curriculum Content	**Hours**
Unit I: Nature of Environmental Studies: Definition, scope and importance, Significance of environmental studies, Multidisciplinary nature of environmental studies. Its need for public awareness.	05
Unit II: Natural resources and associated problems: a) Forest resources: Use and over-exploitation, deforestation, Timber extraction, mining, dams and their effects on forests and tribal people. b) Water resources: Use and over-utilization of surface and ground water, floods, drought, conflicts over water, dams-benefits and problems. c) Mineral resources: Usage and exploitation, environmental effects of extracting and using mineral resources. d) Food resources: World food problem, changes caused by agriculture effects of modern agriculture, fertilizer-pesticide problems. e) Energy resources: Growing energy needs, renewable and non-renewable energy sources, use of alternate energy sources. f) Land resources: Land as a resource, land degradation, man induced landslides, soil erosion and desertification. g) Role of an individual in conservation of natural resources. h) Equitable use of resources for sustainable lifestyle.	08
Unit III : Ecosystems: Concept of an ecosystem, Structure and function of an ecosystem, Producers, consumers and decomposers, Energy flow in the ecosystem, Ecological succession, Food chains, food webs and ecological pyramids. Introduction, types, characteristics features, structure	08

and function of the following Ecosystem: a) Forest ecosystem b) Grassland ecosystem c) Desert ecosystem d) Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries)	
Unit IV: Biodiversity and its conservation: Introduction – Definition: genetic, species and ecosystem diversity, Bio geographical classification of India, Value of biodiversity: consumptive use, productive use, social, ethical, aesthetic and option values.; Biodiversity at global, National and local levels.; India as a mega-diversity nation; Western Ghats as a bio-diversity region; Hot-spots of biodiversity; Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts; Endangered and endemic species of India; Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.	09

Suggested Text Books:	
1.	Agarwal, K. C. 2001, Environmental Biology, Nidi Publ. Ltd., Bikaner.
2.	BharuchaErach, The Biodiversity of India, Mapin Publishing Pvt. Ltd., Ahmedabad, 380013, India
3.	Brunner R. C., 1989, Hazardous Waste Incineration, McGraw Hill Inc. 480p
Suggested Reference Books:	
1.	Clark R. S., Marine Pollution, Clanderson Press Oxford (TB) Pg No. 6
2.	Cunningham, W. P. Cooper, T. H. Gorhani, E. & Hepworth, M. T. 2001, Environmental Encyclopedia, Jaico Publ. House, Mumbai, 1196p
3.	De A. K., Environmental Chemistry, Wiley Eastern Ltd.
4.	Down to Earth, Centre for Science and Environment (R)
5.	Gleick, H., 1993, Water in crisis, Pacific Institute for Studies in Dev., Environment & Security. Stockholm Env. Institute. Oxford Univ. Press 473p
6.	Hawkins R. e., Encyclopedia of Indian Natural History, Bombay Natural History Society, Bombay (R)
7.	Heywood, V. H. & Watson, R. T. 1995, Global Biodiversity Assessment, Cambridge Univ. Press 1140p.
8.	Jadhav, H. &Bhosale, V. M. 1995, Environmental Protection and Laws, Himalaya Pub. House, Delhi, 284p.
9.	Mckinney, M. L. &Schocl. R. M. 1996, Environmental Science Systems & Solutions, Web enhanced edition
10.	Mhskar A. K., Matter Hazardous, Techno-Science Publications (TB)
11.	Miller T. G. Jr., Environmental Science, Wadsworth Publishing Co. (TB)
12.	Odum, E. P. 1971, Fundamentals of Ecology, W. B. Saunders Co. USA, 574p.
13.	Rao M. N. &Datta, A. K. 1987, Waste Water Treatment, Oxford & IBH Publ. Co. Pvt. Ltd.,
14.	Sharma B. K., 2001, Environmental Chemistry, Goel Publ. House, Meerut
15.	Survey of the Environment, The Hindu (M)
16.	Townsend C., Harper, J. and Michael Begon, Essentials of Ecology, Blackwell Science (TB)
17.	Trivedi R. K., Handbook of Environmental Laws, Rules, Guidelines, Compliances and Standards, Vol. I and II, Enviro Media (R)
18.	Trivedi R. K. and P. K. Goel, Introduction to air pollution Techno-Science Publications (TB)
19.	Wagner K. D., 1998, Environmental Management, W. B. Saunders Co. Philadelphia, USA.

Class, Part & Semester	**:**	**Second Year B. Tech (Computer Science and Technology), Part II, & Sem - IV**								
Course Title	**:**	**Introduction to Performing Arts**						***Course Code***	**:**	HS212
Teaching Scheme (Hours)	**:**	2 Hours /Week= 2 x13= 26 hours						***Total Credits***	**:**	Nil
Evaluation Scheme (Marks)	**:**	Assignments Viva voce	: :	50 25	Written Test Grand Total	: :	25 100	***Duration of SEE***	**:**	NA
Revision:	**:**	Fourth						***Month***	**:**	June 2021
Pre-requisites *(if any)*	**:**	No pre-requisite as such is needed however students' involvement and interest in the classroom will make it more lively activity.								
Course Domain	**:**	Humanities and Arts								

Course Rationale : Performing arts are an important part of our lives, our communication and our self-expression. These arts encourage learners to explore their emotions, expanding their imagination and helping them develop their own, unique voice. Each discipline, music, dance and drama, engage their brain, body and emotions in different ways to encourage their confidence and find joy in self-expression. So introducing the learner to such arts may be an interesting experience.

Course Assessment Method:The students will be given five assignments each for 10 marks. At the end of the course, there will be a written test of 25 marks and a viva voce of 25 marks. All these assessments will be for a total of 100 marks. Based on the marks obtained, they will be awarded with a grade similar to other credit courses. Though it is an audit course, obtaining passing grade is essential.

Course Objectives:The Course Teacher will		***Course Outcomes:*** Students will be able to	
1.	State about various performing arts and explain the importance of the same.	1.	Identify the types of performing arts and their differences with importance.
2.	Elucidate about drama, Natya-Shastra etc.	2.	Acquire knowledge about drama, Natya-Shastra, street play etc.
3.	Explain types of dance, will reveal about theaters.	3.	Demonstrate dance skills and organize about theater activities.
4.	Demonstrate about Rag and Taal.	4.	Receive and respond to the Rag and Taal.
5.	List Gharana system and classify Indian musical instruments.	5.	Identify Gharana and instruments of their choice and interest for practice
6.	Summarize contribution of great musicians and outline about music concerts	6.	Recognize contribution of great musicians and display performances for a music concert.

Curriculum Content	**Hours**
Unit I: Introduction to Music, Dance & Drama, History of Indian Music, Various Forms of Vocal Music.	04
Unit II: History and introduction of Drama, Bharat Muni Natya Shastra, street play, Sanskrit Natya, Marathi SangitRangbhumi.	04
Unit III: Dance, its type, Greek and Roman theatres.	04
Unit IV: Concept of Raga, Concept of Taal.	04
Unit V: Notation System, Study of Gharana system in Music, Classification of Indian	05

Instruments, Instrumental Music.	
Unit VI: Contribution of Great Musicians, Appreciation of Music. Performance of a Music Concert.	05
Suggested Reference Books:	
1.	SangeetVisharad, Vasant, SangeetKaryalaya, HatrasPrakashan.
2.	SuchitaBidkar, 'Sangeet Shastra Vigyan', SanskarPrakashan.
3.	SudhirMainkar, 'Sangeet Kala AaniShikshan', SanskarPrakashan.
4.	BhaskarChandavarkar, 'Vadyavedh', SanskarPrakashan.
5.	Arvind Mulgaonkar, 'Tabla', Popular Prakashan.
6.	Chris Hogget,'All about theatre-Off stage'.
7.	MrinaliniSarabhai,'Understanding of Bharat Natyam'.
8.	Joan Borysenko ,'Minding the body and mending the mind'.
9.	V.K.Subbanna,'RagadalliAntrang'.

Class, Part & Semester	**:**	**Second Year B. Tech (Computer Science and Technology), Part II, Sem - IV**					
Course Title	**:**	**Theory of Computation**			***Course Code*:**	**:**	**CS 221**
Teaching Scheme (Hours)	**:**	Lecture : 3Hrs/week Tutorial : 1Hrs/week			***Total Credits***	**:**	4
Evaluation Scheme (Marks)	**:**	CIE=30 (20+10)	SEE = 70	Grand Total=100	***Duration of SEE***	**:**	3 hrs
Revision:	**:**	Fourth			***Month***	**:**	June 2021
Pre-requisites *(if any)*	**:**	Discrete Mathematical Structure					
Course Domain	**:**	Core **(Theory of Computation)**					

Course Rationale:

Theory of Computation deals with the concepts of automata, formal languages, grammar. The reasons to study this course is that the theory provides a simple, elegant view of the complex machine that we call a computer. It is a study of abstract machines that serve as models for computation - finite automata, pushdown automata, and Turing machines and examines the relationship between these automata and formal languages.

Course Objectives: The Course teacher will		***Course Outcomes:*** Students will be able to	
1.	Help to develop ability to understand and conduct mathematical proofs for computation and algorithms.	1.	To analyze problem-solving situations in related areas of theory in computer science.
2.	Introduce students to the mathematical foundations of computation including automata theory.	2.	Enhance ability to understand and conduct mathematical proofs for computation and algorithms.
3.	Demonstrate to design DFA and NFA for solution to engineering problems.	3.	Design deterministic and nondeterministic automata to recognize specified regular languages.
4.	Explain the theory of formal languages and grammars.	4.	Analyze and design finite automata, pushdown automata, formal languages, and grammars.
5.	Demonstrate the PDA and normal forms of grammar.	5.	Convert among equivalently powerful notations for a language, including among DFAs, NFAs, and regular expressions, and between PDAs and CFGs.
6.	Explain different types of Turing Machines.	6.	Design and analyze Turing Machine.

Curriculum Content	**Hours**
Unit 1 Proofs and Regular Languages: Types of Proofs, Mathematical Induction and Recursive definitions with examples. Regular expressions & corresponding regular languages, examples and applications, unions, intersection & complements of regular languages.	6
Unit 2 Finite State Machines: Deterministic finite automata definition and representation, Non-deterministic F.A., NFA with ^ transitions, Equivalence of DFAs, NFAs and NFA-^s. Kleene's theorem - part I & II statements & proofs, minimum state FA for a regular language, minimizing number of states in an FA.	10
Unit 3 Grammars & Languages: Definition and types of grammars and languages, derivation trees and ambiguity, CFL's & Non CFL's., Union, Concatenation and Kleene's operations, Intersection and complements of CFLs, Pumping Lemma & examples.	6
Unit 4 Chomsky Normal Form: BNF and CNF notations, Eliminating ^ production and unit productions from a CFG, Eliminating useless variables from a Context Free Grammar.	3
Unit 5 Push Down Automata: Definition, deterministic PDA, types of acceptance and conversions to each other, CFGs & PDAs., Top-Down, & Bottom-up parsing.	6
Unit 6 Turing Machines: Models of computation, definition of TM as Language Acceptors, Combining Turing machines, computing a function with a TM. Variations in TM, TMs with doubly-infinite tapes, more than one tape, Non-deterministic TM and Universal TM.	8

Suggested list of Tutorials and Assignments:

1. Mathematical Induction
2. Regular expressions & regular languages
3. DFA, NFA and NFA- /\
4. Kleene's theorem - part I & II
5. Grammars and languages
6. Derivation trees and ambiguity
7. Chomsky Normal Form
8. Push Down Automata
9. Top-Down, & Bottom-up parsing
10. Turing Machines
11. Variations in TM

Suggested Text Books:

1.	"Introduction to Languages & Theory of Computation", John C. Martin, TMH, 3rd Edition.
2.	"Discrete Mathematical Structures with Applications to Computer Science", J. P. Tremblay & R. Manohar, Tata McGraw-Hill Edition, 35th Reprint.

Suggested Reference Books:

1.	"Introduction to Automata Theory, Languages and Computations", John E. Hopcraft, Rajeev Motwani, Jeffrey D. Ullman (Pearson Edition).
2.	"Introduction to Theory of Computations", Michael Sipser, Thomson Brooks/Cole.

Class, Part & Semester	**:**	**Second Year B. Tech (Computer Science and Technology), Part II, Sem - IV**					
Course Title	**:**	**Advanced Microprocessor**			***Course Code*:**	**:**	**CS 222**
Teaching Scheme (Hours)	**:**	Lecture : 3Hrs/week Tutorial : 1Hrs/week			***Total Credits***	**:**	**04**
Evaluation Scheme (Marks)	**:**	CIE=30 (20+10)	SEE = 70	Total=100	***Duration of SEE***	**:**	3 hrs
Revision:	**:**	Fourth			***Month***	**:**	June 2021
Pre-requisites *(if any)*	**:**	Basic knowledge of microprocessor,TASM& MASM					
Course Domain	**:**	**Core**					

Course Rationale: The course is aim to understand 8086 family of microprocessor, Programming of 8086 family ,Bus microcontroller , 80386 microprocessor and PIC microcontroller

Course Objectives: The Course teacher will		***Course Outcomes:*** Students will be able to	
1.	Help to analyze the architecture, instruction set and operations of microprocessors 8086 and contemporary peripherals.	**1.**	Get complete knowledge of architecture, instruction sets and operations of microprocessors 8086.
2.	Elaborate the single and multi-processor mode of 8086 processor.	**2.**	Understand 8086 microprocessor, multiprocessor addressing modes.
3.	Discuss to develop assembly level programs for microprocessor and microcontroller.	**3.**	Develop various assembly language programs and understands the various addressing modes required for assembly language programming.
4.	Describe and analyze 80386 microprocessor and PIC microcontroller.	**4.**	Develop enough confidence to take up the challenges in building useful microprocessor based applications.
5.	Illustrate and analyze I/O Interfacing and Interrupt handling concept and to implement these concepts with Intel 8086 Assembly Language.	**5.**	Analyze instruction sets, applying programming and gain hands-on experience of 8086 & 80386 microprocessor and microcontroller.
6.	Elaborate the operation of microprocessors and microcontrollers, machine language programming and interfacing techniques.	**6.**	Outline the architecture of ARM processor and PIC microcontroller.

Curriculum Content	**Hours**
Unit I 8086 CPU Architecture, EU & BIU activities, Segmentation and address transition, 8086 pin description, 8284 clock generation 8286, 8282, configuration of 8086. Accessing even and add address memory with byte/ word. Software and Hardware interrupts.	8
Unit II Addressing modes, data Transfer, arithmetic logical string, i/o instruction, control group ofinstruction, writing programs using assembler directive and in different module and linking, BIOS /DOS interrupts for Printer, VDU, serial, FDC, Add on cards interface.	8
Unit III Multifunction pins of 8086, 8088-Bus controller, IOB mode of 8288, Minimum & Maximum mode Configuration diagram. Study of 8087 NDP	3
Unit IV Linking and relocation, Stacks, procedures, interrupt and interrupt routines, macros, program design, program design examples.	4
Unit V Salient features of 80386DX, Architecture and signal description, Register organization, addressing modes, data types, Real address mode, protected mode, Segmentation, Paging.	5
Unit VI PIC Microcontroller 8 bit Microcontroller, architecture, Addressing Modes, Timers, Counters, Interrupts, Serial Communication, Programming Concepts, design of embedded systems with microcontrollers.	11

Suggested list of Tutorials and Assignments:

1.Explain 8086 CPU architecture with diagram
2.Explain data trans receiver, clock generator, address latches with diagram
3. List and explain addressing modes of 8086
4.Explain with neat sketch diagram bus controller 8088
5. Explain linking and relocation in detail.
6.Explain silent features of 80386
7.Explain with neat sketch diagram PIC microcontroller

General Instructions: Student is evaluated during Continuous Internal Evaluation (Internal Test) and Semester End Examination.

Suggested Text Books:

1.	*"8086/8088 Family design programming and interfacing",* John Uffenbeck, PHI.8th Edition.
2.	*"Design with PIC Microcontrollers",* John B. Peatman, Pearson Education.4th Edition

Suggested Reference Books:

1.	*"The INTEL Microprocessor"*.
2.	*"An introduction to 8086/8088 assembly language programming for beginners",* N. M. Morris.
3	*"Microcomputer Systems: The 8086 / 8088Family Architecture, Programming and Design", Yn - cheng Liu and Gibson, G.A. Prentice Hall of India, 2nd Edition, 2006.*

Class, Part & Semester	**:**	**Second Year B. Tech (Computer Science and Technology) Part II, & Sem IV**					
Course Title	**:**	**Computer Organization**			***Course Code*:**	**:**	**CS 223**
Teaching Scheme (Hours)	**:**	Lecture:	3 Hrs/week		***Total Credits***	**:**	3
		Tutorial:	---				
Evaluation Scheme (Marks)	**:**	CIE=30 (20+10)	SEE = 70	Grand Total=100	***Duration of SEE***	**:**	03 hrs
Revision:	**:**	*Fourth*			***Month***	**:**	*February 2021*
Pre-requisites *(if any)*	**:**	The prerequisite for this course is basic knowledge of digital logic and computer hardware basics.					
Course Domain	**:**	**Core (Logic gates, Boolean algebra, Microprocessors)**					

Course Rationale: Computer architecture and Organization focuses on the function and design of various components necessary to process information digitally. The study of computer architecture and organization focuses on the interface between hardware and software, and emphasizes the structure and behavior of the system.

Course Objectives:The Course teacher will		***Course Outcomes:*** Students will be able to	
1.	Conceptualize the basics of organizational and architectural issues of computer	**1.**	Ability to understand basic structure of computer.
2.	Help to analyze performance issues in processor and memory design of a computer.	**2.**	Ability to perform computer arithmetic operations.
3.	Discuss various data transfer techniques in computer.	**3.**	Ability to understand control unit operations.
4.	Explain to analyze processor performance improvement using instruction level parallelism.	**4.**	Ability to design memory organization that uses banks for different word size operations.
5.	Provide the knowledge on Instruction Level Parallelism.	**5.**	Ability to understand the concept of cache mapping techniques.
6.	Provide the knowledge and Analyze Memory Organization.	**6.**	Ability to understand the concept of I/O organization.
		7.	Ability to conceptualize instruction level parallelism.

Curriculum Content	**Hours**
Unit 1 Basic Computer Organization: Evolution of computers - Mechanical era, Electronic computers, Generations, VLSI era, CPU organization , communications, user and supervisor modes, accumulator based CPU, System bus, instruction cycle, types of instruction(zero, one, two and three address machines), IO interface, RISC & CISC, definition, comparison and examples.	3
Unit 2 CPU design: Specifications, (memory, speed, frequency etc.) with example, Instruction fetching, decoding, executing, Case Study (architecture, block diagram, instruction sets etc.), Pentium 4 processor, AMD	8

processor.	
Unit 3 Computer Arithmetic: Data Representation, basic formats, storage order, fixed point numbers, binary, signed, decimal, hexadecimal, Floating point numbers, basic formats, normalization, biasing, IEEE754 format, Fixed point arithmetic - Addition and subtraction, overflow, high speed adders, adder expansion, Fixed point multiplication - Two's complement multiplier, Booth's algorithm, Combinational array multiplier, Fixed point division - Restoring, Non restoring algorithm, Combinational array divider, Division by repeated multiplication, Floating point arithmetic - Basic operations, Difficulties, Floating point units, Addition, subtraction, multiplication, division.	8
Unit 4 Control Design: Introduction, multi cycle operation, implementation methods, Hardwired control, design methods, state tables, GCD processor, Classical method, one hot method, Design example- twos complement multiplier control, CPU control unit design.	4
Unit 5 Micro programmed control: Basic concepts, control unit organization, parallelism in microinstructions, Microinstruction addressing, timing, Control unit organization, Design example- twos complement, multiplier control, Control field encoding, encoding by function, multiple microinstruction formats.	8
Unit 6 Memory Organization: Types of memory, Memory systems, multilevel, address translation, memory allocation, Caches, Associative memory, direct mapping, set associative addressing.	8
Suggested list of Tutorials and Assignments: NO **General Instructions: ---**	

Suggested Text Books:	
1.	Computer Architecture and Organization - John P Hayes (MGH) 3rd Edition.
2.	Computer Systems Organization & Architecture – John D. Carpinelli (Pearson Education)
Suggested Reference Books:	
1.	Computer Organization - HamacherZaky (MGH).
2.	http://cse.stanford.edu/class/sophomore-college/projects-00/risc/risccisc/ (RISC vs CISC)
3.	http://www.cpu-world.com/sspec/
4.	http://www.intel.com/technology/itj/q12001/pdf/art_2.pdf (The Micro architecture of the Pentium 4 Processor).
5.	http://www.amd.com/usen/assets/content_type/white_papers_and_tech_docs/30579_AMD_Processor_Evaluation_Guide3.1.pdf (AMD Processor Performance Evaluation Guide)

Class, Part & Semester	**:**	**Second Year B. Tech (Computer Science and Technology), Part II, Sem- IV**					
Course Title	**:**	**Software Engineering**			***Course Code*:**	**:**	**CS224**
Teaching Scheme (Hours)	**:**	Lecture :	**3 hours/weeks**		***Total Credits***	**:**	**3**
		Tutorial :	----				
Evaluation Scheme (Marks)	**:**	CIE=30 (20+10)	SEE = 70	Total=100	***Duration of SEE***	**:**	3 hrs
Revision:	**:**	Fourth			***Month***	**:**	June 2021
Pre-requisites *(if any)*	**:**	Fundamental concepts and techniques for analysis, design and implementation of computer programming.					
Course Domain	**:**	Core					

Course Rationale: Software Engineering (SE) comprises the core principles consistent in software construction and maintenance: fundamental software processes and life-cycles, requirements analysis, methodologies and standard notations, principles of software architecture and re-use, software quality frameworks and validation, software development, and maintenance environments and tools. An introduction to object-oriented software development process and design.

Course Objectives: The Course teacher will		***Course Outcomes:*** Students will be able to	
1.	Provide a knowledge of basic Software engineering methods and practices, and their appropriate applications.	**1.**	Apply the project management and analysis principles to S/W project development
2.	Give a general understanding of software process models such as the waterfall and evolutionary models and an understanding of software requirements and the SRS document.	**2.**	Design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability
3.	Elaborate to know role of project management in planning, scheduling, risk management, different software architectural styles, implementation issues such as modularity and coding standards.	**3.**	Identify and solve engineering problems and to gain Knowledge about software development life cycle.
4.	Provide a knowledge of software testing approaches such as unit testing and integration testing and understanding of software evolution and related issues such as version management.	**4.**	Communicate effectively and the broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context
5.	Illustarte quality control and how to	**5.**	Apply the design & testing principles to S/W

	ensure good quality software.		project development to maintain software systems.
6.	Explain some ethical and professional issues that are important for software engineers and development of significant teamwork and project based experience.	**6.**	Identify and Apply methods for software quality and its control.

Curriculum Content	**Hours**
Unit I: Introduction & Software Processes: The S/W problem, S/W applications, the S/W Engineering Approach & Benefits. Software Process, Characteristics of a software process, Software development process, project management process, Software configuration management process, process management process.	6
Unit II: S/W requirements Engineering & Planning Software Project: S/W requirements, problem Analysis, Requirements Specification, validation, metrics. Project Management Plan, a. Cost estimation, project scheduling, staffing and personnel planning. b. Software Configuration Management plans, Quality Assurance plans. c. Project Monitoring Plans, Risk Management.	7
Unit III: Software Design: Objective, Design principles, module level concepts, Design notation and specifications, Artifacts system design document & detailed design document, Structured Design methodology, Verification, Metrics.	6
Unit IV: Object Oriented Design with UML: Introduction , Object Technologies, Unified Process, Use Case Model : methods, Use Case View - Use Case diagrams, Activity diagrams, Design Views- Class Diagrams, Interaction Diagrams, State Chart Diagrams, Process View- Class and Interaction Diagrams, Implementation View- Component Diagrams, Deployment View- Deployment Diagrams. (Case studies on UML views for business requirements).	7
Unit V: Coding & Software Testing: Programming Practice, verification, Metrics: Testing Fundamentals, Testing Levels, Functional testing, Structural testing, Testing object oriented programs, Regression Testing, Testing process Metrics-Reliability Estimation.	8
Unit VI: Software Quality & Project Monitoring and Control: Objectives, need for improvement, cost of Quality, Software quality factors, Total Quality Management, Quality standards such as ISO, CMM and CMMI along with their comparison, Six Sigma Project Manager Skills, Team management, Project tracking, milestone analysis, Activity – level Analysis using SPC, Defect Analysis and prevention, Process monitoring and Audit.	5

Suggested list of Assignments:

1: To understand software characteristics, attributes of a good software and Software Processes.

2: To prepare SRS, problem Analysis, and Project Management Plan.

3: To understand Design principles, Design notation and specifications, Structured Design

methodology.

4: To understand Use Case View, Design Views, Process View, Implementation View, Deployment View.

5: To understand the strategies for software testing.

6: To understand the Importance of quality assurance and ISO 9000 quality standards.

Suggested Text Books:	
1.	*"An integrated approach to S/W engineering",* Pankaj Jalote, Narosa Publishers, 3rd Edition.
2.	*"Software Project Management in practice",* Pankaj Jalote, Pearson Education.
Suggested Reference Books:	
1.	*"Software Engineering: Practitioner's Approach",* Roger S. Pressman, 6th Edition
2.	*"Software Engineering",* Jawadekar W.S, TMGH, 6th Edition
3.	Software Engineering by Kogent Wiley India Limited.
4.	*"Managing Software Engineering: CASE studies and solutions",* Gillies A.C. and Smith P, Chapman and Hall, London.
5.	*"Object oriented software concepts",* Bertrand Mayer.

Class, Part & Semester	**:**	**Second Year B. Tech (Computer Science and Technology), Part II, Sem - IV**					
Course Title	**:**	**Applied Mathematics – II**			***Course Code*:**	**:**	**CS225**
Teaching Scheme (Hours)	**:**	Lecture : 3 Hrs/week Tutorial : 1 Hrs/week			***Total Credits***	**:**	04
Evaluation Scheme (Marks)	**:**	CIE=30 (20+10)	SEE = 70	Grand Total=100	***Duration of SEE***	**:**	3 hrs
Revision:	**:**	Fourth			***Month***	**:**	June 2021
Pre-requisites *(if any)*	**:**	Basic knowledge of Engineering Mathematics-I, Engineering Mathematics-II and Applied Mathematics-I					
Course Domain	**:**	Basic Sciences					

Course Rationale:

This course offers a mathematical understanding for engineering applications. This course produce graduates with mathematical knowledge, computational skills and the ability to deploy these skills effectively in the solution of problems, principally in the area of engineering.

Course Objectives: The Course teacher will		***Course Outcomes:*** Students will be able to	
1.	Elaborate numerical methods and statistics.	**1.**	Understand the difficulty of solving problems analytically and the need to use numerical approximations for their resolution.
2.	Analyze engineering problems based on probability	**2.**	Apply numerical methods for solving problems in different areas of engineering.
3.	Familiarize with correlation and regression.	**3.**	Gain the basic knowledge of correlation and regression.
4.	Provide a knowledge of the Test of Hypotheses and Significance.	**4.**	Formulate and solve different problems in the field computer engineering using probability and test of Significance.
5.	Discuss and solve Transportation Problem.	**5.**	Analyze and solve engineering problems using transportation problem.
		6.	Formulate a mathematical model for engineering problem, solve and interpret the solution in real world.

Curriculum Content	**Hours**
Unit I Numerical solution of algebraic and transcendental equations Zeroes of polynomial and transcendental equation using Bisection method, Iterative method, Secant method, Regula-falsi method and Newton-Raphson method, Newton-Raphson method for system of equations, Mullers method, Rate of convergence of above methods.	7
Unit II Interpolation, Numerical Differentiation and Numerical Integration Lagrange's interpolation formula, Newton's forward and backward difference interpolation	6

formula, Newton's divided difference interpolation formula, Numerical differentiation based on interpolation, Numerical Integration: Trapezoidal Rule, Simpson's 1/3 rd rule, Simpson's 3/8 th rule.	
Unit III Curve Fitting Fitting of Curves by method of Least-squares for linear, parabolic, and exponential, Coefficient of correlation, Spearman's rank correlation, coefficient and lines of regression of bivariate data.	7
Unit IV Probability Random variable, Mean, median, mode and standard deviation. Binomial, Poisson, and Normal distributions.	6
Unit V Test of Significance Sampling distribution of mean and standard error, Large sample tests: Test for an assumed mean and equality of two population means. Small sample tests: t-test for an assumed mean and equality of means of two populations, Paired t-test. Test by using Chi – square distribution. Goodness of fit test. Test for independence of attributes Yates's Correction.	7
Unit VI Transportation Problem Introduction, Mathematical formulation, Method for obtaining initial basic feasible solution, North –West corner method, Low cost entry method, Vogel′s approximation method, Method to obtain optimal solution (MODI Method).	6

Suggested list of Tutorials and Assignments:

1. Zeroes of algebraic and transcendental equations
2. Examples on interpolation
3. Examples on numerical differentiation and integration
4. Examples on correlation and curve fitting
5. Examples on regression.
6. Statistical distributions
7. Examples on Chi – square test.
8. Transportation problem.

General Instructions:

1. Batch wise tutorials are to be conducted. The number of students per batch should be as per the practical batches.
2. Students must be encouraged to solve engineering mathematics problems using different software's in tutorial class only.
3. Each Student has to write at least 6 assignments on entire syllabus

Suggested Text Books:

1.	M. K. Jain, S. R. K. Iyengar, R. K. Jain, "Numerical methods for scientific and Engineering Computation" , 2012, New Age International Limited Publishers.
2.	S. C. Gupta and V. K. Kapoor, "Fundamentals of Mathematical Statistics", 2020.

3.	B. S. Grewal, "Higher Engineering Mathematics", 43rd Edition , Khanna Publishers, Delhi.
4.	S. D. Sharma, "Operations Research ", 11th Edition.
Suggested Reference Books:	
1.	S.C. Chapra, R.P. Canale, "Numerical method for Engineers", 2015, Tata McGraw Hill Publications
2.	James L. Johnon, "Probability and Statistics for Computer science", 2011.
3.	H. K. Dass, "Advanced Engineering Mathematics", 2014, S. Chand Publishing.
4.	Erwin Kreyszig, "Advanced Engineering Mathematics", Fifth Edition , John Wiley & Sons.
5.	M. D. Greenberg, "Advanced Engineering Mathematics", 2nd Edition , Pearson Education.
6.	C. R. Wylie, "Advanced Engineering Mathematics", 6th Edition , McGraw Hill Publication, New Delhi.

Class, Part& Semester	**:**	**Second Year B. Tech (Computer Science and Technology), Part II, Sem-IV**					
Course Title	**:**	**Advanced Microprocessor Lab**			***Course Code*:**	**:**	**CS 222L**
Teaching Scheme (Hours)	**:**	Practical :	**2**Hrs/week		***Total Credits***	**:**	1
Evaluation Scheme (Marks)	**:**	IPE=50	EPE=50	Total=100	***Duration of EPE***	**:**	**03 hours**
Revision:	**:**	Fourth			***Month***	**:**	June 2021
Pre-requisites *(if any)*	**:**	Study of 8085, Assembly language programming for 8085, Writing subroutine, Designing & implementing hardware for INTR and Study of interfacing.					
Course Domain	**:**	**Core**					

Course Rationale: This course will provide an opportunity to the students to become familiar with 8086 microprocessor architecture, instruction set and programming using software MASAM/TASAM.

Course Objectives:The Course teacher will		***Course Outcomes:*** Students will be able to	
1.	Explain 8086 microprocessor, knowledge of 8086 instruction set and ability to utilize it in assembly language programming	**1.**	Apply the knowledge of the fundamentals of assembly level programming of microprocessors and microcontroller.
2.	Illustate Assembly language programming using MASM (Micro Assembler).	**2.**	Learn MASM assembler programming.
3.	Develop the conceptual understanding of 8086 ALP and conduct experiments on data processing.	**3.**	Understand an ALP in 8086 and its interfacing circuits.
4.	Elaborate assembly language programming using 8051 microcontroller.	**4.**	Develop ability in designing a microprocessor and microcontroller systems.
5	Develop ability in programming using microprocessors and microcontrollers.	**5.**	Provide practical hands-on experience with microprocessor applications and interfacing techniques.
6	Explain real mode Memory addressing and ability to interface various devices to the microprocessor.	**6**	Understand and familiarizing with the assembly level programming and microprocessor and microcontroller.

List of Experiments

Sr. No.	**Name of Experiment**
1.	8086 Architecture: To understand 8086 Architecture in details.
2.	Implement 8086 program for addition and subtraction of two 16 bit numbers.

3.	Implement 8086 program for signed and unsigned multiplication.
4.	Implement 8086 program for signed and unsigned division.
5.	Implement 8086 program to check number is even or odd.
6.	Implement 8086 program for check number is positive or negative.
7.	Implement a program: a)To find largest number from array. b)To find smallest number from array.
8.	Implement program for password matching.
9.	Implement a program to display a string and to do case conversion.
10.	Implement a program to string reverse and string copy.
11.	Implement a program: a)To sort numbers in ascending order. b)To sort numbers in descending order.
12.	Implement a program for counting 1's and 0's.
13.	Write NDP architecture in detail with diagram.
General Instructions: Practical Journal Assessment, Internal Oral Examination and External Practical Examination	
Suggested Text Books/ Reference Books/Manual	
1.	*"8086/8088 Family design programming and interfacing",* John Uffenbeck, PHI.2nd Edition
2.	*"An introduction to 8086/8088 assembly language programming for beginners",* N. M. Morris.

Class, Part & Semester	**:**	**Second Year B. Tech (Computer Science and Technology), Part II, Sem IV**					
Course Title	**:**	**Linux and Shell Programming**			***Course Code*:**	**:**	**CS 226L**
Teaching Scheme (Hours)	**:**	Lecture : 2Hrs/week Practical : 2Hrs/week			***Total Credits***	**:**	3
Evaluation Scheme (Marks)	**:**	IPE=50	EPE=50	Total=100	***Duration of EPE***	**:**	3 hrs
Revision:	**:**	Fourth			***Month***	**:**	June 2021
Pre-requisites *(if any)*	**:**	---					
Course Domain	**:**	**Core(Operating System)**					

Course Rationale:
This course gives the hands on experience of the open source operating system approach to programming.

Course Objectives: The Course teacher will		***Course Outcomes:*** Students will be able to	
1.	Familiarize students with the Linux environment	**1.**	Work confidently in Linux environment
2.	Teach the Vi editor at an introductory level of proficiency	**2.**	Ability to use the Vi editor
3.	Familiarize students the fundamentals of shell scripting/programming	**3.**	Ability to use Shell Programming using Linux
4.	Help to perform simple concurrent programs	**4.**	Ability to write Shell Programming using Linux
5.	Explain to write and use moderately complex regular expressions	**5.**	Ability to Write and use moderately complex regular expressions.
6.	Familiarize students with basic Linux administration.	**6.**	Ability to perform basic Linux administration

Curriculum Content	**Hours**
Unit I Introduction to Linux and Linux utilities – A brief history of Linux Architecture , Features of Linux , Linux commands- PATH, man, echo, printf, script, passwd, uname, who, date, sty, pwd, cd, mkdir, rmdir etc.	2
Unit II The File SystemBasic File Attributes, the vi Editor	2
Unit III The Shell, The Process, Customizing the environment	2
Unit IV	

More file attributes, Simple filters	2
Unit V Filters using regular expressions	3
Unit VI Essential Shell Programming, awk – An Advanced Filter	3

List of *Experiments* and Assignments:

1. Basic Shell Commands

Shell Programs:

1. Fibonacci Series
2. Designing Calculator
3. File Operations
4. Base conversion
5. Usage of cut and grep commands
6. Usage of user defined functions

Administration

Root, The administrators Privileges, Maintaining Security, user management, Managing Disk Space , Device Files

General Instructions: Students have to perform minimum 8 practicals

Suggested Text Books:

1.	Unix Concepts and Applications, 4thedititon , Sumitabha Das , MGH
2.	Linux system programming, Robert Love, O` Reilly, SPD

Suggested Reference Books/Manual

1.	Beginning Linux Programming, 4th edition, N . Mathew, R.stone, Wrox Willey India Edition
2.	Linux, The Complete Reference , 6th edition, Richard Petersen, MGH

Class, Part& Semester	**:**	**Second Year B. Tech (Computer Science and Technology), Part II, Sem- IV**					
Course Title	**:**	**Object Oriented Programming Lab**			***Course Code*:**	**:**	**CS 227L**
Teaching Scheme (Hours)	**:**	Theory : **2 hrs /week** Practical: **2 hrs /week**			***Total Credits***	**:**	**3**
Evaluation Scheme (Marks)	**:**	IPE =**50**	EPE =**50**	Total=100	***Duration of EPE***	**:**	03 Hours
Revision:	**:**	Fourth			***Month***	**:**	June 2021
Pre-requisites *(if any)*	**:**	Knowledge of Programming Methodology, 'C' language, Control Statements, Functions, Arrays, Pointers, Structures and Union and File Handling concepts.					
Course Domain	**:**	Core					

Course Rationale: This course is an advanced level programming course using the C++ language.

Course Objectives: The Course teacher will		***Course Outcomes:*** Students will be able to	
1.	Explain the basic concepts and techniques which form the object oriented programming paradigm.	**1.**	Explain what constitutes an object-oriented approach to programming and identify potential benefits ofobject-oriented programming over other approaches.
2.	Strengthen their problem solving ability by applying the characteristics of an object-oriented approach.	**2.**	Apply an object-oriented approach to developing applications of varying complexities.
3.	Introduce object oriented concepts in C++.	**3.**	Take a problem and develop the structures to represent objects and the algorithms to perform operations.
4.	Elaborate fundamentals of programming such as variables, conditional and iterative execution, methods, etc.	**4.**	Apply standards and principles to write truly readable code.
5	Help to implement the object oriented concepts to solve problems	**5.**	Test a program and, if necessary, find mistakes in the program and correct them.
6	Demonstrate to develop an application applying the object oriented concepts	**6**	Develop applications using object oriented concepts.

Curriculum Content	**Hours**
Unit I: Introduction to Object Oriented Programming: Introduction to procedural, object-oriented programming, Limitations of procedural programming, Need of object-oriented programming, fundamentals of object-oriented programming: objects, classes, data members, methods, messages, data encapsulation,	3 Hrs

data abstraction and information hiding, inheritance, polymorphism.	
Unit II: Basics of C++ programming: Variable declarations, global scope, const variables, reference variables, function prototypes, functions with default arguments, call by value, call by reference, returning by reference, call by pointer, inline functions, constant arguments, 'cin', 'cout', formatting and I/O manipulators, Classes and Objects defining Class, data members, member functions, Access specifiers – public, private, protected, constructor, destructor, array of objects, passing objects to functions, returning object.	6 Hrs
Unit III: Inheritance: Need of Inheritance, Concept, public, private, protected inheritance, Single inheritance, Multiple and multilevel inheritance, Hybrid Inheritance, Virtual base class, overriding of member functions, static variable, static function, friend function, friend class	4 Hrs
Unit IV: Polymorphism: Pointers basics of memory management, New and delete operators, Pointer to object, Pointer to data members, this pointer. Need of Polymorphism, concept, Compile time polymorphism or early binding: function over loading and operator overloading, operator overloading using member function and friend function, overloading - unary, binary, arithmetic operators, relational operators, Overloading new and delete operators, insertion and extraction operators, Run time polymorphism or late binding using Virtual function, pure virtual function, Abstract class, Type conversion	5 Hrs
Unit V: Files and Streams: Concept of Streams, concept of File, opening and closing a file, detecting end-of-file, file modes, file pointer, reading and writing characters, strings and objects to the file, operations to move file pointers i.eseekg, seekp, tellg, tellp.	4 Hrs
Unit VI: Advanced C++ features: Introduction to Generic Programming using Templates: Function template and class template, Introduction to Standard Template Library (STL), containers, iterators and algorithms, study of container template classes for vectors and stacks and related algorithms Exception handling: Introduction, syntax for exception handling code: try-catch-throw, Multiple Exceptions, Exceptions with arguments	4 Hrs

List of Experiments

Sr. No.	Name of the Experiment
1.	Write a program to demonstrate concept of class. For example: create class matrix, class string, class car, class date, class time, class person etc.
2.	Write a program to demonstrate following Function concepts a. Function overloading b. Constructors of all types c. Default parameters, returning by reference d. Demonstration of friend function e. Demonstration of static function
3.	Write a program to demonstrate a. Operator overloading –for unary as well as binary operation. b. Apply above concept on matrix and string classes created above.

4.	Write a program to demonstrate C^{++} s capability of all types of inheritance a. Single, multiple, multivalued b. Virtual function. c. Abstract class d. Runtime polymorphism
5.	Write a program for new and delete operators, pointers to objects.
6.	Write a program for pointers to pointers, this pointer.
7.	Write a program for Templates, Exception handling.
8.	Write a program for Stack and Queue.
9.	Write a program for the linked list,
10.	Write a program for Binary tree, Traversal of a Binary tree.
Suggested Text Books/ Reference Books/Manual	
1.	C++: The Complete Reference Fourth Edition - Herbert Schildt (McGraw-Hill) , 4th edition
2.	C++ programming: From Problem Analysis to Program Design Fifth Edition -D.S. Malik (Cengage Learning)
3.	C++ Programming with language –Bjarne Stroustrup (AT & T), 4th edition
4.	Object Oriented Programming with C++ Fourth Edition-E Balguruswamy (McGraw-Hill), 4th edition
5.	Object oriented Programming in C++ 3rd Edition-R.Lafore (Galgotia Publications), 3rd Edition
6.	C++ programming –John Thomas Berry(PHI), 2nd Edition
7.	Object –Oriented Analysis & Design: Understanding System Development with UML 2.0 , Docherty, Wiley India Ltd.
8.	http://www.spoken-tutorial.org/ NMEICT Project of Govt. Of India.

Class, Part & Semester	**:**	**Second Year B. Tech (Computer Science and Technology), Part II & Sem- IV**					
Course Title	**:**	**Environmental Studies**			***Course Code***	**:**	HS211
Teaching Scheme (Hours)	**:**	Lecture : 02 Hour/Week Tutorial : 00 Hours/Week			***Total Credits***	**:**	Nil
Evaluation Scheme (Marks)	**:**	CIE = 00 SEE = 70	IPE=30 Project	: Grand Total=100	***Duration of SEE***	**:**	3 hrs. At the year end
Revision	**:**	Fourth			***Month***	**:**	June 2021
Pre-requisites *(if any)*	**:**	HS211					
Course Domain	**:**	Ethics and Environment					

Course Rationale: The Course is all about learning the way we should live and how we can develop sustainable strategies to protect the environment. It helps individuals to develop an understanding of living and physical environment and how to resolve challenging environmental issues affecting nature.

Course Objectives: The Course Teacher will		***Course Outcomes:*** Students will be able to	
1.	Explain the types of environmental pollution.	1.	Identify the pollutants and respond to the pollution problem
2	Help to make the students recognize social issues and the environment connectivity with the same.	2.	Acquire knowledge of ecological threats and choose for sustainable developments.
3.	Discuss various environmental Protection Acts reveal the students the importance of the same.	3.	Anticipate all these laws and follow the same for the care of the environment.
4.	Explain the students to adapt to various environmental technologies.	4.	Apply their knowledge to implement pollution prevention measure through some practical work.

Curriculum Content	**Hours**
Unit V Environmental pollution: Definition: Causes, effects and control measures of: a) Air pollution, b) Water pollution, c) Soil pollution, d) Marine pollution, e) Noise pollution, f) Thermal pollution, g) Nuclear hazards Solid waste Management: Causes, effects and control measures of urban and industrial wastes. Role of an individual in prevention of pollution. Pollution case studies Disaster management: Floods, earthquake, cyclone and landslides. Tsunami	06
Unit VI Social issues and the environment : From Unsustainable to Sustainable development; Urban problems related to energy; Water conservation, rain water harvesting, watershed management; Resettlement and rehabilitation of people; its problems and concerns; Environmental ethics: Issue and possible solutions; Climate change, Global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust; Wasteland reclamation; Consumerism and waste products.	08
Unit VII Environmental protection : Environment Protection Act.; Air (Prevention and Control of Pollution) Act.; Water (Prevention and control of Pollution) Act; Wildlife Protection Act; Forest Conservation Act; Population Growth and Human Health, Human Rights. ;Field	06

Work-Visit to a local area to document environmental assets river/forest/grassland/hill/mountain or Visit to a local polluted siteurban/rural/Industrial/Agricultural or Study of common plants, insects, birds or Study of simple ecosystems-ponds, river, hill slopes, etc.	
Unit VIII Project / Field work:	10

Suggested Text Books:	
1.	Agarwal, K. C. 2001, Environmental Biology, Nidi Publ. Ltd., Bikaner.
2.	BharuchaErach, The Biodiversity of India, Mapin Publishing Pvt. Ltd., Ahmedabad, 380013, India
3.	Brunner R. C., 1989, Hazardous Waste Incineration, McGraw Hill Inc. 480p
Suggested Reference Books:	
1.	Clark R. S., Marine Pollution, Clanderson Press Oxford (TB) Pg No. 6
2.	Cunningham, W. P. Cooper, T. H. Gorhani, E. & Hepworth, M. T. 2001, Environmental Encyclopedia, Jaico Publ. House, Mumbai, 1196p
3.	De A. K., Environmental Chemistry, Wiley Eastern Ltd.
4.	Down to Earth, Centre for Science and Environment (R)
5.	Gleick, H., 1993, Water in crisis, Pacific Institute for Studies in Dev., Environment & Security. Stockholm Env. Institute. Oxford Univ. Press 473p
6.	Hawkins R. e., Encyclopedia of Indian Natural History, Bombay Natural History Society, Bombay (R)
7.	Heywood, V. H. & Watson, R. T. 1995, Global Biodiversity Assessment, Cambridge Univ. Press 1140p.
8.	Jadhav, H. &Bhosale, V. M. 1995, Environmental Protection and Laws, Himalaya Pub. House, Delhi, 284p.
9.	Mckinney, M. L. &Schocl. R. M. 1996, Environmental Science Systems & Solutions, Web enhanced edition
10.	Mhskar A. K., Matter Hazardous, Techno-Science Publications (TB)
11.	Miller T. G. Jr., Environmental Science, Wadsworth Publishing Co. (TB)
12.	Odum, E. P. 1971, Fundamentals of Ecology, W. B. Saunders Co. USA, 574p.
13.	Rao M. N. &Datta, A. K. 1987, Waste Water Treatment, Oxford & IBH Publ. Co. Pvt. Ltd.,
14.	Sharma B. K., 2001, Environmental Chemistry, Goel Publ. House, Meerut
15.	Survey of the Environment, The Hindu (M)
16.	Townsend C., Harper, J. and Michael Begon, Essentials of Ecology, Blackwell Science (TB)
17.	Trivedi R. K., Handbook of Environmental Laws, Rules, Guidelines, Compliances and Standards, Vol. I and II, Enviro Media (R)
18.	Trivedi R. K. and P. K. Goel, Introduction to air pollution Techno-Science Publications (TB)
19.	Wagner K. D., 1998, Environmental Management, W. B. Saunders Co. Philadelphia, USA.

Class, Part & Semester	**:**	**Second Year B. Tech (Computer Science and Technology), Part II & Sem- IV**								
Course Title	**:**	**Soft Skills Development**						***Course Code***	**:**	HS222
Teaching Scheme (Hours)	**:**	2 Hours /Week= 2 x13= 26 hours						***Total Credits***	**:**	Nil
Evaluation Scheme (Marks)	**:**	Assignments Viva voce	: :	50 25	Written Test Grand Total	: :	25 100	***Duration of SEE***	**:**	NA
Revision	**:**	Fourth						***Month***	**:**	June 2021
Pre-requisites *(if any)*	**:**	H. S. C. Level English language competency								
Course Domain	**:**	Humanity and Arts								

Course Rationale: The course skills focus on who people are, as opposed to what they are trained in. These skills serve to represent learners' approach to life and work. The course develops interpersonal skills hardwired to an individual's personality, and such skills characterize how we interact with other people in the workplace. These skills are important because they enable students to adjust to the frustrations and challenges they will encounter in their adult life, as well as the demands of work. Mastering soft skills help students learn, live and work better.

Course Assessment Method: The students will be given five assignments each for 10 marks. At the end of the course, there will be a written test of 25 marks and a viva voce of 25 marks. All these assessments will be for a total of 100 marks. Based on the marks obtained, they will be awarded with a grade similar to other credit courses. Though it is an audit course, obtaining passing grade is essential.

Course Objectives: The Course Teacher will		***Course Outcomes:*** Students will be able to	
1.	Illustrate the components of self-development and state the importance of career planning.	1.	Identify components of self-development and realize its importance in their career planning.
2.	Define Communication and classify the same.	2.	Differentiate between different communication types and apply the same.
3.	Explain behavioral skills, team skills and interpersonal skills.	3.	Acquire behavioral, team and interpersonal skills and display the same.
4.	Explain to classify documentation types and describe various types of report writing.	4.	Follow different document formats and acquire report and proposal writing skills.
5.	Describe emotional intelligence and its role.	5.	Receive and respond to emotions with intelligence.
6.	Paraphrase interview skills and demonstrate resume writing.	6.	Acquire interview skills and apply those when required.

Curriculum Content	**Hours**
Unit I Self Development: Self-analysis, creativity, attitude, motivation, goal setting. Importance of career visioning and planning.	02
Unit II Effective Communication Skills: Importance of communication, Communication process, Elements of communication, Communication Types-verbal and non-verbal, objectives of communication. Business Communication, current English usage, debates, language games,	06

situational dialogues, precise writing, essay writing, presentations.	
Unit III Behavioral Skills: Psychological Tests: Aptitude and personality assessment, suggestions for improvement, **Team Skills:** Team building and leadership, evolution of groups into teams, group dynamics, emergence of leadership, intra-group dynamics, inter-group dynamics, conflict management, inter dependency, assessment of team-based projects, **Time Management:** Pareto's Principle, Parkinson's Laws, Murphy's Laws, Law of Clutter, prioritization, goal setting, effective time management, **Interpersonal Skills:** Negotiations, listening skills, social skills, assertive skills, cross-cultural communications, **Leadership Skills:** Concepts of leadership, leadership styles, insights from great leaders.	08
Unit IV Documentation: Report writing-Formal report, study tour report, project report, Writing proposal-solicited proposals and unsolicited proposals.	03
Unit V Emotional Intelligence: Emotional Brain, Nature of emotional intelligence, emotional intelligence applied windows of opportunity, emotional literacy.	04
Unit VI Interview Skills: Importance of Interview Skills, Resume Building, Group discussion and personal interview, Psychometric Test, actual career planning.	03

Suggested Text Books:	
1.	Soft Skills, 2015, Career Development Centre, Green Pearl Publications.
Suggested Reference Books:	
1.	Seven Habits of Highly Effective Teens, Covey Sean, New York, Fireside Publishers, 1998.
2.	How to win Friends and Influence People, Carnegie Dale, New York: Simon & Schuster, 1998.
3.	I am ok, You are ok ,Thomas A Harris, New York-Harper and Row, 1972
4.	Emotional Intelligence, Daniel Goleman, Bantam Book, 2006
5.	Effective communication skill, MTD training &Ventus publishing ApS ISBN 978-87-7681-598-1.

Equivalence of Second Year B.Tech (Computer Science and Technology) Semester III and IV

The above detailed syllabus is a revised version of the Second Year B. Tech (**Computer Science and Technology**) Program being conducted by the Shivaji University at the Technology Department of the University. This syllabus is to be implemented from June 2021, (Academic year 2021-22). The prime feature of this revision is the transformation of the existing curriculum into the Outcome based curriculum as specified in NBA rules and regulations.

The Equivalence for the subjects/coursesof **Computer Science and Technology** at Second Year B Tech Semester IIIand IV pre-revised Program under the faculty of Engineering and Technology is as follows.

Second Year B.Tech Semester III (Computer Science and Technology)

Sr.No	Second Year B.Tech(Computer Science and Technology) Semester III Pre-revised syllabus	Second Year B.Tech(Computer Science and Technology) Semester III Revised syllabus	Remark
1	Engineering Mathematics-III	Applied Mathematics-I	Course Name Changed Slight modification in the content
2	Discrete Mathematical Structure	Discrete Mathematical Structure	No change in the subject content
3	Digital Systems and Microprocessor	Digital System and Microprocessor	No change in the subject content
4	Data Structures with C	Data Structures	Course Name Changed Slight modification in the content
5	Data Communication	Data Communication and Networking	Course Name Changed Slight modification in the content
6	Digital System and Microprocessor Lab	Digital System and Microprocessor Lab	No change in the subject content
7	Data Structures Lab	Data Structures Lab	No change in the subject content
8	Unix and Shell Programming	Data Communication and Networking Lab	Unix and Shell Programming Shifted to Semester-IV and with changed name Linux and Shell Programming.
9	Environmental Studies	Environmental Studies	No change in the subject content
10	Introduction to Performing Arts	Introduction to Performing Arts	No change in the Audit Subject content

Second Year B.Tech Semester IV (Computer Science and Technology)

Sr.No	Second Year B.Tech(Computer Science and Technology) Semester IV Pre-revised syllabus	Second Year B.Tech(Computer Science and Technology) Semester IV Revised syllabus	Remark
1.	Theory of Computation	Theory of Computation	No change in the subject content
2.	Advanced Microprocessor	Advanced Microprocessor	Slight modification in the content
3.	Computer Organization	Computer Organization	Slight modification in the content
4.	Computer Networks	Software Engineering	Computer Networks Shifted to Semester-III with subject name Data Communication and Networking
5.	Computational Mathematics	Applied Mathematics-II	Course Name Changed Slight modification in the content
6.	Advanced Microprocessor Lab	Advanced Microprocessor Lab	No change in the subject content
7.	Computer Networks Lab	Linux and Shell Programming Lab	Computer Networks Lab Shifted to Semester-III with subject name Data Communication and Networking Lab
8.	Object Oriented Lab	Object Oriented Programming Lab	Course Name Changed Slight modification in the content
9.	Environmental Studies Project Work	Environmental Studies	No change in the subject content
10.	Soft Skills Development	Soft Skills Development	No change in the subject content

Audit course have not been assigned any credits. The students will be evaluated for these courses by the concerned course in charge. There will be grade conferred to the student. The grade will be based on conversion of marks obtained out of 50. (Obtaining passing grade is essential). Please refer to chart in the detail examination scheme. The chart shows the marks range and the respective grade.