B. Tech. Computer Science & Engineering (Second Year) Academic Year: 2018 -19
Trimester – IV

Sr. No.	Course Code	Name of Course	Category	Weekly Workload, Hrs			Credits		Assessment Marks**			
				Theory	Tutorial	Lab	Th	Lab	CCA	LCA	ETT	Total
1	CS211	Discrete Structures	BS	3	1	--	3	--	100	-	50	150
2	CS212	Principles of Programming Languages	PC	3	-	2	2	1	50	50	50	150
3	CS213	Digital Electronics and Logic Design	PC	3	-	2	2	1	50	50	50	150
4	CS214	Object Oriented Programming	PC	3	-	2	2	1	50	50	50	150
5	CS215	Computer Organization	PC	3	-	-	2	-	50	-	50	100
6	ES	Environmental Science	HSS	2	-	-	1	-	50	-	-	50
		Total :		17	01	06	12	03	350	150	250	750

**Assessment Marks are valid only if Attendance criteria are met

B. Tech. Computer Science & Engineering (Second Year) Academic Year: 2018 -19

Trimester – V

Sr. No.	Course Code	Name of Course	Category	Weekly Workload, Hrs			Credits		Assessment Marks**			
				Theory	Tutorial	Lab	Th	Lab	CCA	LCA	ETT	Total
1	ES222	Mathematics -III	BS	3	1	--	3	--	100	-	50	150
2	CS221	Data Structures - I	PC	3	-	2	2	1	50	50	50	150
3	CS222	Microprocessor and Interfacing Techniques	PC	3	-	2	2	1	50	50	50	150
4	CS223	Data and Mobile Communication	PC	3	-	2	2	1	50	50	50	150
5		Science and Spirituality	WP	3	-	-	2	-	50	-	50	100
6		National Study Tour	WP	-	-	-	-	-	-	-	-	-
		Total :		15	1	06	11	03	300	150	250	700

**Assessment Marks are valid only if Attendance criteria are met

Dr. Vishwanath Karad
MIT WORLD PEACE UNIVERSITY | PUNE
TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

B. Tech. Computer Science & Engineering (Second Year) Academic Year: 2018 -19
Trimester – VI

Sr. No.	Course Code	Name of Course	Category	Weekly Workload, Hrs			Credits		Assessment Marks**			
				Theory	Tutorial	Lab	Th	Lab	CCA	LCA	ETT	Total
1	CS231	Data Structures - II	PC	3	-	2	2	1	50	50	50	150
2	CS232	Microprocessor and Microcontroller	PC	3	-	2	2	1	50	50	50	150
3	CS233	Software Engineering and Project Management	PC	3	-	2	2	1	50	50	50	150
4	CS234	Operating Systems	PC	3	-	2	2	1	50	50	50	150
5	IC	Indian Constitution	HSS	2	-	-	1	-	50	-	-	50
		Total :		14	-	08	09	04	250	200	200	650

**Assessment Marks are valid only if Attendance criteria are met

Dr. Vishwanath Karad
MIT WORLD PEACE UNIVERSITY | PUNE
TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

COURSE STRUCTURE

Course Code	**CS211**			
Course Category	**Basic Science**			
Course Title	**Discrete Structures**			
Teaching Scheme and Credits	**L**	**T**	**Laboratory**	**Credits**
Weekly load hrs	**3**	**1**	**--**	**2+1=3**

Pre-requisites:

- Basic Mathematics

Course Objectives:

1. To understand the basic concept of Set theory, Relations and Functions.
2. To study discrete mathematic concept in computer algorithms and programming languages.
3. To learn the concept of Tree and Graph theory to solve real-world computer science problems.

Course Outcomes:

After completion of this course students will be able to:

1. Design logic to formulate and solve a problem using concept of Set theory, Relations and Functions.
2. Develop logic for solving problems in computer science.
3. Solve computer science problem by applying the concept of Tree and Graph theory.

Course Contents:

1) Set Theory
2) Counting
3) Relations
4) Functions
5) Graph
6) Trees

Tutorial:

1) Problem Solving on Set Theory
2) Questions on Counting
3) N-ary and Equivalence Relations
4) Problems on Bijective and Recursive Functions
5) Adjacency matrix and Shortest path problems using Graph
6) Huffman and Binary Search Trees

Learning Resources:

Text Books:

1. Kenneth H. Rosen, ―Discrete Mathematics and its Applications, Tata McGraw-Hill, ISBN 978-0-07-288008-3, 7th Edition.
2. C. L. Liu, ―Elements of Discrete Mathematics, TMH, ISBN 10:0-07-066913-9.

Reference Books:

1. Bernard Kolman, Robert C. Busby and Sharon Ross, ―Discrete Mathematical Structures, Prentice-Hall of India /Pearson, ISBN: 0132078457, 9780132078450.
2. Dr. K. D. Joshi, ― Foundations of Discrete Mathematics, New Age International Limited, Publishers, January 1996, ISBN: 8122408265, 9788122408263

Supplementary Reading:

1. N. Biggs,"Discrete Mathematics", 2nd Edition, Oxford University Press
2. Data Structures – Seymour Lipschutz, Shaum's outlines, MCGraw – Hill Inc.

Web Resources:

https://learn.saylor.org/course/cs202
https://www.mooc-list.com/tags/discrete-mathematics

Web links:

https://www.tutorialspoint.com/discrete_mathematics/index.htm

MOOCs:

http://nptel.ac.in/courses/106106094/3
https://www.coursera.org/learn/discrete-mathematics

Pedagogy:

- Chalk and Board
- PPT
- Two Teacher Method
- Video Lectures

Assessment Scheme:

Class Continuous Assessment (CCA): 100 Marks (100%)

Assignments	Test	Tutorials	MCQ	Attendance
-	30%	30%	20%	20%

Term End Examination: 50 Marks (100% weightage)

Syllabus:

Unit No.	Contents
1	**Set Theory:** Sets, Combinations of sets, Venn Diagrams, Finite and Infinite sets, Uncountable infinite sets, Principle of inclusion and exclusion, Multisets, Cartesian Product and Power Set. **Counting:** The Basics of Counting, Permutations and Combinations, Binomial Coefficients, Algorithms for generating Permutations and Combinations, The Pigeonhole Principle.
2	**Relations :** Relations and Their Properties, n-ary Relations and Their Applications, Representing Relations , Closures of Relations, Equivalence Relations, Partial Orderings- Chain, Anti chain and Lattice, Groups, Types of Groups. **Functions:** Subjective, Injective and Bijective functions, Inverse Functions and Compositions of Functions, Recursive Function.
3	**Graphs:** Graph and Graph Models, Graph Terminology and Types of Graph, Representing Graph and Graph Isomorphism, Connectivity, Euler and Hamilton Paths, Single source shortest path- Dijkstra's pseudo code algorithm, Planar Graph.
4	**Trees:** Introduction, Properties of trees, Binary search tree, Decision tree, Prefix codes and Huffman coding, Spanning Trees and Minimum Spanning Tree - Kruskal‘s and Prim’s pseudo code algorithms , Case Study- Game Tree.

Tutorial List:

A. Set Theory:

1) Draw a Venn diagram for the symmetric difference of the sets A and B.
2) Let A, B and C be sets. Show that
   a. $(A \cup B) \subseteq (A \cup B \cup C)$
   b. $(A - B) - C \subseteq A - C$

B. Counting:

1) Find the expansion of $(x + y)^4$
   a. Using Combinatorial Reasoning
   b. Using Binomial Theorem
2) How many different ways are there to choose a dozen donuts from the 21 varieties at a donut shop?

C. Relations:

1) The 4-tuples in a 4-ary relation represents these attributes of published book: title, ISBN, Publication Date, Number of Pages.
   a. What is a likely primary key for this relation?
   b. Under what conditions would (title, publication date) be a composite key?
   c. Under what conditions would (title, Number of Pages) be a composite key?

2) Which of these relations on the set of all people are equivalence relations?
   a. { (a, b) | a and b are the same age}
   b. { (a, b) | a and b have the same parents}
   c. { (a, b) | a and b share a common parent}

D. Functions

1) Show that the following functions are primitive recursive
   a. Exponentiation
   b. Factorial Function
2) Determine whether each of these functions is a bijection from R to R
   a. $f(x)=2x+1$
   b. $f(x)=x^2+1$

E. Graph

1) Draw a graph with following adjacency matrix

1	1	0
1	0	1
0	1	0

2) Is a shortest path between two vertices in a weighted graph unique if the weights of edges are distinct?

F. Trees

1) Build a binary search tree for the words banana, peach, apple, pear, coconut, mango and papaya using alphabetical order.
2) Use Huffman coding to encode these symbols with given frequencies: a: 0.20, b: 0.10, c: 0.15, d: 0.25, e: 0.30. What is the average number of bits required to encode a character?

COURSE STRUCTURE

Course Code	**CS212**			
Course Category	**Professional Core**			
Course Title	**Principles of Programming Languages**			
Teaching Scheme and Credits	**L**	**T**	**Laboratory**	**Credits**
Weekly load hrs	**3**	**-**	**2**	**2+1= 3**

Pre-requisites

- Computer Science and Information Technology-I
- Computer Science and Information Technology-II

Course Objectives:

1. To learn programming language features and designs
2. To analyze and compare different programming paradigms of languages
3. To gain overall knowledge of programming languages

Course Outcomes:

After completion of this course, students will be able to:

1. Apply appropriate programming paradigm in real-time application
2. Select appropriate programming language for solving problems

Course Contents:

1. Introduction
2. Basic Building Blocks of a Language
3. Procedures
4. Functional Programming Languages (FPL)

Laboratory Exercises / Practicals:

1. Basic functional programming language command execution
2. Implement Factorial program using lisp
3. Implement bubble sort using lisp
4. Case study of Haskell

Learning Resources:

Text Books:

1. Pratt T.W, Zelkowitz "Programming Languages: Design and Implementation" PHI, 2002, 4th Edition.
2. Sethi Ravi, "Programming Languages: Concepts and Constructs" Addison Wesley1996.

Reference Books:

1. Sebesta R. W, 'Concepts of programming languages', Pearson Education 2001, 4th edition.
2. Carlo Ghezzi, Mehdi Jazayeri, 'Programming Language Concepts', 3rd edition, Wiley Publications.
3. Patric Henry Winston and Berthold Klaus Paul Horn, 'LISP', Pearson Education, 3rd edition.

Supplementary Reading:
https://www.cs.rutgers.edu/~lou/314-f04-slides/topic01_intro.post.pdf

Web Resources:

https://www.khanacademy.org/computing/computer-science/
https://www.hackerrank.com/contests

Web Links:
www.nptel.ac.in/course.php
https://videoken.com
https://www.tutorialspoint.com

MOOCs:
https://www.cs.cmu.edu/~rwh/courses/ppl/

Pedagogy:

- Power Point presentations
- White board teaching
- Few video lectures (ex. NPTEL)

Assessment Scheme:

Class Continuous Assessment (CCA) : 50 Marks

Assignments	Test	Tutorial	MCQ	Attendance
20%	30%	-	30%	20%

Laboratory Continuous Assessment (LCA)-50 Marks

Practical	Oral based on practical	Site Visit	Mini Project	Problem based Learning	Any other (Mock 1 and Mock2)
60%	-	-	-	-	40%

Term End Examination: 50 Marks (100% weightage)

Syllabus :

Unit No.	Contents
1	**Introduction** Characteristics of Programming Languages, Influencing Factors for the Evolution of Programming Language, Desirable Features and Design Issues. **Brief Introduction to Programming Language Paradigms:** Imperative, Object Oriented, Functional, Logic and Concurrent Programming. **Syntactic Structure:** Syntax, Semantics, Structure, Character Set Tokens, Sentence-Syntax and Semantics, Expression Notation, Grammar, Syntax Tree, Context Free Grammar, Translators.
2	**Basic Building Blocks of a Language** **Data Representation:** Data Object, Declaration of Variables, Constants, Data Types, Properties of Structured and Non-Structured Data Types. **Data Types**: Derived and Abstract Data Types, Type Checking, Binding and Binding Times, Type Conversion, Control Flow Statements. **Storage Structure:** Implementation and Storage Representation of Data Types.
3	**Procedures** **Subprogram:** Simple Call Subprogram, Recursive Subprogram, Static And Dynamic Scope, Referencing Environment (Local, Non-Local and Global). **Parameter Passing**: Parameter Passing Methods, Dynamic Scope of Variables. **Activation Records:** Control Flow between Activations, Elements of Activation Records.

4	**Functional Programming Languages (FPL)** **Introduction to FPL:** Fundamentals and Elements of FPL, Function Declaration, Expression Evaluation, Application of Functional Programming Languages, Comparison of Functional and Imperative Languages. **Type Checking:** Type Inference, Type Names and Type Equivalence, Coercion. **Case Studies**: LISP, Haskel.

Laboratory:

Sr. No.	Contents	Lab Hours
1	Basic functional programming language command execution	02
2	Implement Factorial program using lisp	02
3	Implement bubble sort using lisp	02
4	Implementation of Basic python commands.	02
5	Implement Bubble sort using python	02
6	Implement Factorial program using python	02
7	Case study of Haskell	02

COURSE STRUCTURE

Course Code	**CS213**			
Course Category	**Professional Core**			
Course Title	**Digital Electronics and Logic Design**			
Teaching Scheme and Credits	**L**	**T**	**Laboratory**	**Credits**
Weekly load hrs	**3**	**-**	**2**	**2+1 = 3**

- **Pre-requisites**: Introduction to Electronics Engineering

Course Objectives:

1. To gain knowledge of characteristics of Logic Families
2. To design Combinational and Sequential Circuits
3. To realize concept of ASM chart.
4. To design and implement combinational and sequential circuits
5. To design and implement digital circuits using Very High Speed Integrated Circuit Hardware Description Language (VHDL).

Course Outcomes:

1. Design combinational circuits using gates and MSI Integrated Circuits (ICs).
2. Design sequential circuits using Flip flops and Registers.
3. Devise applications using ASM chart and concept of Moore, Mealy machine.
4. Implement combinational and sequential circuits using Integrated Circuits.
5. Implement combinational and sequential circuits using VHDL program.

Course Contents

1. Combinational Logic Design.
2. Sequential Logic Design.
3. Synchronous Sequential Circuit Design.
4. Introduction to Very High Speed Integrated Circuit Hardware Description Language (VHDL).

Laboratory Exercises / Practicals:

1. To Realize Half Subtractor, Full Subtractor using Basic Gates and Universal gates
2. Realize following functions using IC 74153
   a) Half adder b) 8:1 Mux (Σ m (0,2,4,7))
3. Design and implement 3 bit up and 3 bit down Ripple Counter using IC7476.
4. Design and implement 3 bit Synchronous Up Counter using IC7476.
5. Draw ASM chart for real life digital system (e.g. Washing Machine, Vending Machine, Elevator control, ATM machine)
6. Design & simulate 4 bit up/down synchronous counter using Behavioral modeling in VHDL
7. Design & simulate 4:1 Multiplexer using Dataflow modeling
8. Design & simulate Full adder using structural modeling.

Learning Resources:

Text Books:

1. R.P. Jain, "Modern Digital Electronics", McGraw Hill Education,2012, ISBN–13: 978-0-07- 066911-6., 4th edition
2. J. Bhaskar , "A VHDL Primer", Pearson Education, 2008, ISBN: 9788177582000, 3rd Edition

Reference Books:

1. Morris Mano, "Digital Logic & Computer Design", Pearson Education
2. D. Leach, Malvino, Saha, "Digital Principles and Applications", Tata McGraw Hill, 2011, ISBN – 13:978-0-07-014170-4. , 7th edition
3. Thomas Floyd, "Digital Fundamentals", Pearson Education, 2003, ISBN: 8178088762, 8th Edition

Web Resources:

https://viden.io/knowledge/114/attachments/427?name=viden-K105-Digital+Logic+And+Computer+Design+By+M.+Morris+Mano+%282nd+Edition%29.pdf

Web Links: http://nptel.ac.in/courses/117105080/40

MOOCs: http://nptel.ac.in/courses/117105080/3

Pedagogy:

- Power Point presentations
- White board teaching

Assessment Scheme:

Class Continuous Assessment (CCA) 50 Marks

Assignments	Test	Tutorial	MCQ	Attendance
20%	30%	-	30%	20%

Laboratory Continuous Assessment (LCA) 50 Marks

Practical	Oral based on practical	Site Visit	Mini Project	Problem based Learning	Any other (Mock 1 and Mock2)
60%	-	-	-	-	40%

Term End Examination: 50 Marks (100% weightage)

Syllabus:

Unit No.	Contents
1	**Combinational Logic Design** Characteristics of Digital ICs: Speed, Power dissipation, Figure of merit, Fan-in, Fan-out, Current and voltage parameters, Noise immunity, Operating temperature range. Minimization of SOP and POS equations using K-Maps up to 4 variables. Multiplexers, MUX (IC 74153), MUX tree, De-multiplexers, Decoder (IC74138). Implementation of Half- Adder, Full Adder using MUX and Decoder.
2	**Sequential Logic Design** SR, JK, MS JK, D, T Flip Flops. Truth Tables and Excitation tables. Conversion from one type to another type of Flip Flop. Study of IC7476. Counters: 3 bit up, 3bit down Asynchronous and Synchronous counter. Study of Modulus n counter, IC 7490 and its application to implement MOD Counters. Registers: Buffer register, shift register types- SISO, SIPO, PISO, PIPO. Ring counters, Johnson Counter.
3	**Synchronous Sequential Circuit Design** State diagram, State Tables, State assignment. Moore and Mealy Machine representations, Sequence detector using Mealy model. Algorithmic State Machines (ASM): Finite State Machines (FSM). ASM charts, notations, construction of ASM chart (e.g. Counters, sequence detector for Moore and Mealy model).

4	**Introduction to Very High Speed Integrated Circuit Hardware Description Language (VHDL)** Introduction to VHDL, Library, Packages, Entity, Architecture, Data Objects (Variables, Signals, and Constants), Data Types, Data Operators and VHDL Modeling styles with Programming Examples – Dataflow, Behavioral and Structural. Comparison of modeling styles. Concurrent Statements (When..Else), Sequential statements (If..Else, Loop) Comparison of Concurrent and Sequential statements.

Laboratory:-

Sr. No	Contents	Lab Hours
1	To Realize Half Subtractor, Full Subtractor using Basic Gates and Universal gates.	02
2	Realize following functions using IC 74153 a)Half adder b) 8:1 Mux (Σ m (0,2,4,7))	02
3	Design and implement 3 bit up and 3 bit down Ripple Counter using IC7476.	02
4	Design and implement 3 bit Synchronous Up Counter using IC7476.	02
5	Design & simulate 4 bit up/down synchronous counter using Behavioral modeling in VHDL	02
6	Design & simulate 4:1 Multiplexer using Dataflow modeling in VHDL	02
7	Draw ASM chart for real life digital system (e.g. Washing Machine, Vending Machine, Elevator control, ATM machine)	04
8	Design & simulate Full adder using structural modeling in VHDL.	04

COURSE STRUCTURE

Course Code	**CS214**			
Course Category	**Professional Core**			
Course Title	**Object Oriented Programming**			
Teaching Scheme and Credits	**L**	**T**	**Laboratory**	**Credits**
Weekly load hrs	3	--	2	**2+1= 3**

Pre-requisites:

- Computer Science and Information Technology I and II

Course Objectives:

1. Understand basic concepts of Object Oriented Programming.
2. Learn Inheritance, Polymorphism and Exception Handling features of Object Oriented Programming.
3. Study concepts of Standard Template Library.

Course Outcomes:

After completion of this course, students will be able to:

1. Apply the basic concepts of Object Oriented Programming in application development.
2. Design and develop real world applications using inheritance, Polymorphism and Exception Handling features.
3. Explore and use Standard Template Library to simplify programming.

Course Contents:

1. Introduction to Object Oriented Programming
2. Polymorphism and Inheritance
3. File and Exception Handling
4. Templates and Standard Template Library (STL)

Laboratory Exercises / Practicals:

1. Classes and operator overloading
2. Inheritance and Polymorphism
3. Exception handling
4. File handling
5. Templates
6. Standard Template Library

Learning Resources:

Reference Books:

1. Herbert Schildt, 'C++ The Complete Reference', Fourth Edition, McGraw Hill Professional, 2011, ISBN-13: 978-0072226805
2. Robert Lafore, 'Object-Oriented Programming in C++', Fourth Edition, Sams

Publishing, ISBN: 0672323087, ISBN-13: 978-8131722824

3. Bjarne Stroustrup, 'The C++ Programming language', Third Edition, Pearson Education. ISBN: 9788131705216
4. K. R. Venugopal, Rajkumar Buyya, T. Ravishankar, 'Mastering C++', Tata McGraw-Hill, ISBN 13: 9780074634547

Supplementary Reading:

1. Power Point Slides
2. Lab Manual
3. Question Bank
4. Practice Assignments

Web Resources:

http://ocw.mit.edu

Web Links:

https://nptel.ac.in/courses/106105151/
http://nptel.ac.in/syllabus/106106110/
http://ocw.mit.edu

MOOCs:

https://www.coursera.org/learn/c-plus-plus-a
https://www.mooc-list.com/tags/c-1

Pedagogy:

- Power Point Presentations
- Practical Demos
- Videos
- Expert lectures
- Workshop
- Co Teacher Scheme

Assessment Scheme:

Class Continuous Assessment (CCA) : 50 Marks

Assignments	Test	Tutorial	MCQ	Attendance
20%	30%	-	30%	20%

Laboratory Continuous Assessment (LCA)-50 Marks

Practical	Oral based on practical	Site Visit	Mini Project	Problem based Learning	Any other (Mock 1 and Mock2)
60%	-	-	-	-	40%

Term End Examination: 50 Marks (100% weightage)

Syllabus:

Unit No.	Contents
1	**Introduction to Object Oriented Programming (OOP)** **Fundamentals of OOP:** Class, Object, Data Abstraction, Data Encapsulation, Inheritance, Polymorphism. **Classes and Objects:** Creation, Accessing Class Members, Access Specifiers, Member Function Definition, Inline Function, Static Data Members, Function Overloading, Constructor and Destructor, Objects and Memory Requirements, Friend Functions.
2	**Polymorphism and Inheritance** **Operator Overloading**: Concept of Operator Overloading, Overloading Unary and Binary Operators, Prefix and Postfix Operator Implementation. **Inheritance:** Introduction, Base and Derived Classes, Member Access Control, Inheriting Constructors and Destructors, Types of Inheritance, Overriding Member Functions, Ambiguity in Multiple Inheritance, Virtual Base Class. **Polymorphism and Virtual Functions:** Pointers to Objects, Pointers to Derived Class, Importance of Virtual Function, Pure Virtual Function, Abstract Class, Virtual Destructors, Early and Late Binding.

3	**File and Exception Handling** **File Handling:** Stream and Files, Stream Classes, File Pointers, File I/O with Member Functions, Formatted I/O and I/O Manipulators, Error handling during file operations, Overloading Insertion and Extraction Operators. **Exception Handling**: Introduction, Exception Handling Mechanism - try, catch and throw, Multiple Exceptions, Re-throwing an exception, Exception and Inheritance.
4	**Templates and Standard Template Library(STL)** **Templates**: Function Template, Class Template. Introduction to STL, Containers - Sequence Containers and Associative Containers, Container Adapters, Algorithms and Iterators - input, output, forward, bidirectional and random access.

Laboratory:

Sr.No.	Contents	Lab Hours
1	**Define a class Complex consisting following:** Data members: a. real b. imaginary part Member Functions: a. One default constructor b. Function getdata () to take the value of real and imaginary part. c. Function show() to display and 1. Four overloaded operator member functions i. Operator+ to add two complex numbers. ii. Operator * to multiply two complex numbers. iii. Overloaded << and >> to print and read Complex Numbers. 2. Friend function to add two complex number by taking two reference variables of class complex and returning another reference.	04

2	A company pays its employees on a weekly basis. The employees are of four types: Salaried employees are paid a fixed weekly salary regardless of the number of hours worked, hourly employees are paid by the hour and receive overtime pay for all hours worked in excess of 40 hours, commission employees are paid a percentage of their sales and salaried-commission employees receive a base salary plus a percentage of their sales. For the current pay period, the company has decided to reward salaried-commission employees by adding 10% to their base salaries. The company wants to implement an OO application that performs its payroll calculations polymorphically.	04
3	**Define a class Employee consisting following:** Data members: a. Employee ID b. Name of Employee c. Age d. Income e. City f. Vehicle Member Functions: a. To assign initial values. b. To display. **Accept Employee ID, Name, Age, Income, City and Vehicle from the user. Create an exception to check the following conditions and throw an exception if the condition does not meet.** 1. Employee age between 18 and 55 2. Employee income between Rs. 50,000 – Rs. 1,00,000 per month 3. Employee staying in Pune/ Mumbai/ Bangalore / Chennai 4. Employee having 4-wheeler	02
4	A shop maintains an inventory of items. It stores information of items like Item_Code, Item_Name, Quantity and Cost in a data file. Whenever Customer wants to buy an item, sales person inputs the Item_Code and/or Item_Name and the system searches in a file and displays whether it is available or not otherwise an appropriate message is displayed. If it is, then the system displays the item details and request for the quantity of items required. If the	04

<table>
<tr>
<td></td>
<td>requested quantities of items are available, the total cost of items is displayed; otherwise the message is displayed as required items not in stock. After purchasing an item, system updates the file.

Design a system using a class called Items with suitable data members and member functions. Implement C++ program for the inventory system that will create a data file containing the Record of Items in the following form:

<table>
<tr><th>Item_Code</th><th>Item_name</th><th>Quantity</th><th>Cost in Rs.</th></tr>
<tr><td>3</td><td>Pens</td><td>24</td><td>10</td></tr>
<tr><td>17</td><td>Notebooks</td><td>46</td><td>14.99</td></tr>
</table>

Data Members:
a) Item_Code
b) Item_Name
c) Quantity
d) Cost

Member Function:
a) Create file and store Record of Items
b) Search an Item in the file by Item_Code or Item_Name
c) Arrange the Items by Item_Code or Item_Name
d) Update the file</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Design a class Template to implement stack of integers consisting following member functions:

a) Create
b) Display
c) Push
d) Pop</td>
<td>02</td>
</tr>
<tr>
<td>6</td>
<td>A shop maintains the inventory of items. It stores information of items like Item_Code, Item_Name, Quantity and Cost of it in a list of STL. Whenever Customer wants to buy an item, sales person inputs the Item_Code and/or Item_Name and the system searches in a file and displays whether it is available or not otherwise an appropriate message is displayed. If it is, then the system displays the item details and request for the quantity of items required. If the requested quantity of items are available, the total cost of items is displayed; otherwise the message is displayed as required items not in stock. After purchasing an item, system updates the list.</td>
<td>04</td>
</tr>
</table>

	Design a system using a class called Items with suitable data members and member functions. Implement menu driven C++ program for the inventory system using STL list. **Data Members:** a) Item_Code b) Item_Name c) Quantity d) Cost **Member Function:** a) Create STL list and store Record of Items b) Search an Item in the file by Item_Code or Item_Name(Searching) c) Arrange the Items by Item_Code or Item_Name(Sorting) d) Update the file(Insert and delete).	

COURSE STRUCTURE

Course Code	**CS215**			
Course Category	**Professional Core**			
Course Title	**Computer Organization**			
Teaching Scheme and Credits	**L**	**T**	**Laboratory**	**Credits**
Weekly load hrs	**3**	**--**	**--**	**2**

Pre-requisites:

- Introduction to Electronics Engineering
- Digital Electronics and Logic Design

Course Objectives:

1. To acquire the knowledge of structure, function and evolution of computer systems.
2. To understand instruction level parallelism and internal processor organization.
3. To gain the conceptual knowledge of Cache memory and multiple processor organization.
4. To design Arithmetic Logical Unit and Control Unit of digital computers.

Course Outcomes:

After completion of this course, students will be able to:-

1. Demonstrate computer architecture concepts related to design of modern processors and compare various generations of processors.
2. Design arithmetic functional units such as: Adder, Subtractor, and Multiplier and Division units.
3. Obtain the knowledge of processor structure and its functions for internal designing of processor organization.
4. Design the size of the cache for the various processor organizations.

Course Contents:

1. Computer Function, Interconnections and Evolution
2. Computer Arithmetic
3. Processor Organization and Control Unit
4. Memory and Parallel Processor Organizations

Learning Resources:

Reference Books:

1. W. Stallings, "Computer Organization and Architecture: Designing for performance", Pearson Education/ Prentice Hall of India, 2003, ISBN 978-93-325-1870-4, 7th Edition.
2. Zaky S, Hamacher, "Computer Organization", 5th Edition, McGraw-Hill Publications, 2001, ISBN- 978-1-25-900537-5, 5th Edition.

Supplementary Books:

1. M. Usha, T. S. Srikanth, "Computer System Architecture and Organization", Wiley, ISBN: 978-81-265-2284-2, 2014.
2. A. S. Tanenbaum "Structured Computer Organization", 4th Edition, Prentice Hall of India, ISBN: 81–203–1553–7, 1991,
3. J. Hays, "Computer Architecture and Organization", 2nd Edition, McGraw-Hill, ISBN 0–07–100479–3, 1988

Web Resources:

Web Links:
http://ebooks.lpude.in/computer_application/bca/term_4/DCAP206_INTRODUCTION_TO_COMPUTER_ORGANIZATION_AND_ARCHITECTURE_DCAP502_COMPUTER_ORGANIZATION_AND_ARCHITECTURE.pdf

MOOCs:

https://onlinecourses.nptel.ac.in/noc17_cs35
https://onlinecourses.nptel.ac.in/noc17_cs19

Pedagogy:

- Power point slides
- Chalk and Talk

Assessment Scheme:
Class Continuous Assessment (CCA): 50 Marks

Assignments	Test	Tutorials	MCQ	Attendance
20%	30%	-	30%	20%

Term End Examination: 50 Marks (100% weightage)

Syllabus:

Unit No.	Contents
1	**Computer Function, Interconnections and Evolution** Computer Organization and Architecture, Structure and Function, A brief history of computer, Evolution of Intel microprocessor from 4004 to core i7, Comparison of various generations of microprocessor, Computer components and functions ,Characteristics of Reduced Instruction Set Architectures, CISC versus RISC Characteristics.
2	**Computer Arithmetic** The arithmetic and logic unit, Integer Representation, Integer Arithmetic, Multiplication – Block diagram, Hardware implementation of unsigned binary multiplication, Multiplication of positive numbers, signed number multiplication, Booth's Algorithm, Division – Flowchart for unsigned binary division, Division Algorithms. Floating - Point Representation, IEEE standard.
3	**Processor Organization and Control Unit** Instruction format, Types of Instruction and operations, common addressing techniques, Processor Structure and function - Processor and register organization, Instruction Cycle, Instruction Pipelining, Pipeline Performance, Pipeline Hazards - Structural, Data, Control. Control Unit Operation - The functional requirement of processor, Micro – operation and instruction cycle, Functional Requirements & Operations of the Control Unit, Block diagram of control unit.
4	**Memory and Parallel Processor Organizations** Key characteristics of memory system, memory hierarchy, Cache Memory - Cache memory principles, mapping functions, example of mapping techniques. Multiple Processor Organization: Flynn's Taxonomy, Introduction to Clusters

COURSE STRUCTURE

Course Code	**ES**			
Course Category	**Humanities and Social Sciences**			
Course Title	**Environmental Science**			
Teaching Scheme and Credits	**L**	**T**	**Laboratory**	**Credits**
Weekly load hrs	**2**	**-**	**-**	**1**

Pre-requisites:

Course Objectives:

1) To impart sense of community responsibility by becoming aware of scientific issues in the larger social context.
2) To develop an interdisciplinary approach to complex environmental problems using basic tools of the natural and social sciences including biology chemistry, political sciences and technology.
3) To inculcate ability to work effectively as a member of interdisciplinary team to solve environment related social issues.

Course Outcomes:

After completion of this course students will be able to;

1) Correlate core concepts and methods from ecological and physical sciences and their application in environmental problem solving. CL-II)
2) Reflect critically about their roles and identities as citizens, consumers and environmental actors in a complex, interconnected world.(CL-V)
3) Apply systems, concepts and methodologies to analyze and understand interactions between social and environmental processes (CL-III)

Course Contents:

1. Multidisciplinary nature of environmental science
2. Natural Resources
3. Ecosystem, biodiversity and its conservation
4. Environmental Pollution
5. Social Issues and the Environment

Learning Resources:

Reference Books:

1. Bharucha Erach, The Biodiversity of India, 1st edition Mapin Publishing Pvt.Ltd. Ahmedabad,India,2000.
2. Miller T.G.Jr. Enviornmental Science, 2 nd edition ,Wadsworth Publication1989.

Supplementary Reading:

1.De A.K., Enviornmental Chemistry,7 th edition ,Wiley Eastern Ltd., 2014.

2.Down to Earth- Magazine ,Centre of science and environment,New Delhi, Editor-Sunita Narian

Web Resources:

Weblinks:

https://www.ugc.ac.in/oldpdf/modelcurriculum/env.pdf
http://www.nptel.ac.in/courses/120108005/
http://www.nptel.ac.in/courses/120108004/
http://www.nptel.ac.in/courses/120108002/

MOOCs:

https://www.coursera.org/learn/global-warming

https://www.coursera.org/learn/global-environmental-management

https://www.edx.org/course/climate-change-science-ubcx-climate1x-3
https://www.edx.org/course/sustainable-tourism-society-environmental-aspects

Pedagogy:

- Co-teaching
- Power point presentations
- Videos
- Demonstrations
- Systematic use of group work and project based learning.

Assessment Scheme:

Class Continuous Assessment (CCA): (50 marks)(with % weights)

Assignments	Test	Presentations	MCQ	Attendance and Initiative
60%		40%	Nil	Nil

Syllabus:

Unit No.	Contents
1	**Multidisciplinary nature of environmental science** Definition, scope and importance. Need for public awareness.
2	**Natural Resources** Renewable and non-renewable resources : Natural resources and associated problems. a) Forest resources : b) Water resources c) Mineral resources d) Food resources. e) Energy resources f) Land resources Role of an individual in conservation of natural resources. Case Studies.
3	**Ecosystem,biodiversity and its conservation** Concept, structure, functions and types of an ecosystem .Introduction – Definition of biodiversity: genetic, species and ecosystem diversity. Bio-geographical classification of India. Value of biodiversity. Bio-diversity at global, National and local levels. India as a mega-diversity nation. Hot-spots of biodiversity. Threats to biodiversity. Conservation of bio-diversity .
4	**Environmental Pollution** Definition , Cause, 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 Role of an individual in prevention of pollution. Diaster management : floods, earthquake, cyclone and landslides.
5	**Social Issues and the Environment** From Unsustainable to Sustainable development. Urban problems related to energy . Water conservation, rain water harvesting, watershed management. Resettlement and rahabilitation of people; its problems and concerns. Environmental ethics ,Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust. Wasteland reclamation. Consumerism and waste products. Environmental regulations.Issues involved in enforcement of environmental legislation. Public awareness.

COURSE STRUCTURE

Course Code	**ES222**			
Course Category	**Engineering Sciences**			
Course Title	**Applied Mathematics –III**			
Teaching Scheme and Credits	**L**	**T**	**Laboratory**	**Credits**
Weekly load hrs	**3**	**1**	**--**	**2+1=3**

- **Pre-requisites**: Applied Mathematics-I , Applied Mathematics-II

Course Objectives:

1. To understand integral transform techniques and their applications.
2. To understand the concepts of correlation and regression.
3. To learn various probability distributions.
4. To learn vectors calculus for applications in engineering field.

Course Outcomes:

After completion of this course students will be able to

1. Solve problems related to Fourier and Z transforms.(CL III)
2. Use statistical methodology and tools in the problem-solving process.(CL IV)
3. Use probability models for various engineering problems.(CL IV)
4. Apply the knowledge of vector calculus for solving engineering problems.(CL III)

Course Contents:

1. Transform Techniques
2. Statistics
3. Probability
4. Vector Calculus

Tutorial Exercises:

1. Fourier Sine and Cosine Transforms.
2. Z-Transform and Inverse Z-Transform.
3. Standard deviation, Moments, Skewness, Kurtosis,
4. Correlation and Regression.
5. Bayes theorem, Probability density function and mathematical expectation.
6. Binomial, Poisson and Normal distribution.
7. Vector differentiation, gradient, divergence and curl.
8. Work done, Green's Lemma, Stoke's and Divergence Theorem.

Two tutorials will be conducted using Mathematical Software. Tutorial shall be engaged in four batches (batch size of 15 students) per division.

Learning Resources:

Reference Books

1. Kreyszig Erwin, "Advanced Engineering Mathematics" ,10th edition ,Wiley Eastern Limited 2015.
2. O' Neil Peter, "Advanced Engineering Mathematics" ,8th edition ,Cengage Learning 2015.
3. Greenberg Michael D., "Advanced Engineering Mathematics", 2nd edition, Pearson . 2009.
4. Grewal B.S., "Higher Engineering Mathematics" ,43rd edition Khanna Publishers 2014

Supplementary Reading:

Weber H.J. and Arfken G.B. "Mathematical Methods For Physicists" , 6th edition, Academic Press 2011.

Web Resources:

http://nptel.ac.in/courses/111105035/6
http://nptel.ac.in/courses/111105090

MOOCs :

https://ocw.mit.edu/courses/mathematics/18-02sc-multivariable-calculus-fall-2010/

Pedagogy:

- Co-teaching
- Audio- video techniques
- Tutorials and class tests

Assessment Scheme:

Class Continuous Assessment (CCA): 100 marks

Assignments	Test	Tutorials	MCQ	Attendance
20%	30%	30%	-	20%

Term End Examination: 50 Marks (100% weightage)

Syllabus:

Unit No.	Contents
1	**Transform Techniques:** Fourier Transform: Fourier Integral theorem, Fourier Sine and Cosine Transforms, Inverse Fourier Transform. Z-Transform: Definition, Properties, Z- transform of standard sequences and their inverse, solution of difference equations.
2	**Statistics:** Measures of Central tendency, Dispersion, Moments, Skewness and Kurtosis, Correlation, Method of Least square Linear and multiple linear regression.
3	**Probability**: Basic Probability Theory, Bayes' Theorem, Discrete and Continuous Random Variables, Probability Mass Function, Probability Density Function, Distribution Function, Mathematical Expectation, Probability Distributions, Binomial, Poisson and Normal Distributions.
4	**Vector Calculus**: Vector Differential: Physical interpretation of Vector differentiation, Vector differential operator, Gradient, Divergence and Curl, Directional derivative, Vector identities. Vector Integration: Lines, Surface and Volume, Integration, Work done, Green's Lemma, Stoke's and Divergence Theorem.

COURSE STRUCTURE

Course Code	**CS221**			
Course Category	**Professional Core**			
Course Title	**Data Structures-I**			
Teaching Scheme and Credits	**L**	**T**	**Laboratory**	**Credits**
Weekly load hrs	**3**	**--**	**2**	**2+1=3**

Pre-requisites:

- Computer Science and Information Technology-I
- Computer Science and Information Technology-II
- Discrete Structure

Course Objectives:

1. To realize the concept of Abstract Data Type
2. To study memory representations and operations associated with different data structures
3. To study different sorting and searching methods

Course Outcomes:

After completion of the course the students will be able to :-

1. To select appropriate data structures for problem solving
2. To implement various data structures
3. To compare and analyse different searching and sorting algorithms

Course Contents:

1. Introduction to Data Structures
2. Linear Data Structures
3. Searching
4. Sorting
5. Stacks
6. Queues
7. Linked Lists

Laboratory Exercises / Practicals:

1.Set Operations
2. String Operations
3.Sparse Matrix Operations
4.Searching and Sorting on Student Database
5.Expression Conversion using Stack
6. Implementation of Queue
7. Operations on Singly Linked List
8. Polynomial Operations Using Circular Linked List

Learning Resources

Text Books:

1. Maureen Spankle ,"Problem Solving and Programming Concepts", ISBN: 81-317-0711-E.
2. Horowitz, S. Sahani, S. Anderson-Freed, "Fundamentals of Data Structures in C", Universities Press, 2008

Reference Books:

1. Dennis Ritchie, Kernighan, "The C Programming Language", Prentice Hall.
2. Treamblay, Sorenson, "An introduction to data structures with applications", Tata McGraw Hill, Second Edition.

Supplementary Readings:

1. Aaron Tanenbaum, "Data Structures using C", Pearson Education.
2. R. Gilberg, B. Forouzan, "Data Structures: A pseudo code approach with C", Cenage Learning, ISBN 9788131503140.
3. R.G.Dromy, "How to Solve it by Computers", Prentice Hall.

Web Resources:

https://www.khanacademy.org/computing/computer-science/algorithms
https://www.hackerrank.com/contests/basic-ds-quiz-1/

Web Links:

https://www.tutorialspoint.com/data_structures_algorithms/

MOOCs:

http://nptel.ac.in/courses/106102064/1
https://nptel.ac.in/courses/106103069/

Pedagogy:

- Chalk and Board
- PPT
- Two Teacher Method
- Video Lectures
- Discussion Forum
- Flipped Classroom

Assessment Scheme:

Class Continuous Assessment (CCA)- 50 Marks

Assignments	Test	Tutorial	MCQ	Attendance
20%	30%	-	30%	20%

Laboratory Continuous Assessment (LCA)-100 Marks

Practical	Oral based on practical	Site Visit	Mini Project	Problem based Learning	Any other (Mock 1 and Mock 2)
60%	-	-	-	-	40%

Term End Examination: 50 Marks (100% weightage)

Syllabus:
Theory:-

Unit No.	Contents
1	**Introduction to Data Structures:** Types of problems, Problem solving with computers, Difficulties with problem solving. Data, Data objects, data types, Abstract Data types (ADT) and Data Structure, Types of data structure Introduction to algorithms, Algorithm design tools: Pseudo code and flowchart, Analysis of Algorithms- Space complexity, Time complexity, Asymptotic notations
2	**Linear Data Structures:** Array as an Abstract Data Type, Sequential Organization , Memory Representation and Address Calculation. Representation of Polynomials using arrays , addition and evaluation of Polynomials, Representation of sparse matrix, Addition, Simple Transpose and Fast transpose of sparse

	matrix **Searching:** Linear search, Binary search. **Sorting:** Types of sorting-Internal and External sorting, Sorting methods- Bubble sort, Insertion sort, Selection sort, Merge Sort, Applications of sorting, Comparison and analysis of sorting methods.
3	**Stacks:** Stack as an Abstract Data Type, Representation of Stack Using Sequential Organization, Applications of Stack- Expression Conversion and Evaluation, Recursion. **Queues**: Queue as Abstract Data Type, Representation of Queue Using Sequential Organization , Circular Queue, Advantages of Circular queues, Deque, Application of Queue : Job scheduling.
4	**Linked List:** Linked List as an Abstract Data Type, Representation of Linked List Using Sequential Organization, Representation of Linked List Using Dynamic Organization, Operations on Linked List, Polynomial operations using linked list. Circular Linked List, Doubly Linked List , Generalized Linked List (GLL) **Case Study :** Garbage Collection

Laboratory:-

Sr. No.	Contents	Lab Hours
1	For a second year Computer Engineering class, set A of students like Chocolate Ice-cream and set B of students like butterscotch ice-cream. Write a C program to store two sets using array. Implement following operations- i. Set of students who like either Chocolate or butterscotch or both ii. Set of students who like both Chocolate and butterscotch iii. Set of students who like only Chocolate not butterscotch iv. Set of students who like only butterscotch not Chocolate v. Number of students who like neither Chocolate nor butterscotch	04
2	Write a C program for string operations- copy, concatenate, palindrome, check substring, equal, reverse, frequency count and length (with and without pointers)	04
3	Write a C program for sparse matrix realization and operations on it- Transpose, Fast Transpose and addition of two matrices	06

4	Write a C program to create student database using array of structures. Apply searching and sorting techniques.	04
5	Implement stack as an ADT and apply it for different expression conversions (infix to postfix or infix to prefix, prefix to postfix or prefix to infix, postfix to infix or postfix to prefix).	06
6	Pizza parlor accepting maximum M orders. Orders are served in first come first served basis. Order once placed cannot be cancelled. Write a C program to simulate the system with simple queue using array.[Implement the same system using Circular Queue(Optional)].	04
7	Write a C program to implement Singly Linked List and perform following operations on it. i) Insert a node ii) Delete a node iii) Display linked list iv) Reverse a linked list(Using Recursive Function) v) Revert a linked list(Using Pointers) vi) Sort a list vii) Concatenate two lists	06
8	Implement polynomial operations using Circular Linked List (Create, Display, Addition and Evaluation).	06

COURSE STRUCTURE

Course Code	**CS222**			
Course Category	**Professional Core**			
Course Title	**Microprocessor and Interfacing Techniques**			
Teaching Scheme and Credits	**L**	**T**	**Laboratory**	**Credits**
Weekly load hrs	**3**	**------**	**2**	**2+1=3**

Pre-requisites:

- Digital Electronics and Logic Design
- Computer Organization

Course Objectives:

1. To learn the architecture and programming of 8086 Microprocessor.
2. To learn important peripherals and their interfacing with 8086 Microprocessor.
3. To study working of NDP and Motherboard of IBM PC.

Course Outcomes:

Upon completion of the course, the students will be able to:

1. Understanding: Learn concepts of microprocessor architecture.
2. Applying: Write Assembly Language programs for various applications
3. Understanding: Understand working of motherboard and design systems using the concepts of interfacing

Course Contents:

1. 8086 Architecture
2. Memory and I/O Interfacing
3. Study of Peripherals
4. Math-Controller & Study of Motherboard

Laboratory Exercises / Practical:

1. Character String Display
2. Hex Number Display
3. Array Addition
4. String Operations
5. String Operations using FAR Procedure
6. NDP
7. 8086 DAC interfacing using 8255
8. 8086 interfacing with 8251

Learning Resources:

Text Books:

1. N. Mathivanan,"Microprocessors, PC Hardware and Interfacing", Prentice-Hall India, ISBN-81-203-2317-3.
2. Ray, K. Bhurchandi, "Advanced Microprocessors and peripherals: Arch, Programming & Interfacing", Tata McGraw Hill,2004 ISBN 0-07-463841-6.
3. Introduction to 64 bit Intel Assembly Language Programming for Linux, 2nd Edition, Ray Seyfarth, ISBN10:1478119209, ISBN-13: 9781478119203, 2012.

Reference Books:

1. Douglas Hall, "Microprocessors & Interfacing", McGraw Hill, Revised 2nd Edition, 2006 ISBN 0-07-100462-9

Supplementary Reading:

Web Resources:

http://nptel.ac.in/downloads/106108100/

Web Links:

https://www.tutorialspoint.com/assembly_programming/

MOOCs:

https://www.udemy.com/certificate-program-in-introduction-to-microprocessors/
https://edge.edx.org/courses/course-v1:BITSX+F241+2015-16_Semester_II/2c221095e36f4647b83e662d183c1984/

Pedagogy:

- White Board,
- Power Point Presentations
- Two Teachers method

Assessment Scheme:

Class Continuous Assessment (CCA):50 Marks

Assignments	Test	Tutorial	MCQ	Attendance
20%	30%	-	30%	20%

Laboratory Continuous Assessment (LCA):50 Marks

Practical	Oral based on practical	Site Visit	Mini Project	Problem based Learning	Any other (Mock 1 and Mock2)
60%	-	-	-	-	40%

Term End Examination: 50 Marks (100% weightage)

Syllabus :

Unit No.	Contents
1	**8086 Architecture:** Architecture and Functional Pin description of 8086, Programmers model (Register Set) and Segmentation of 8086, logical to physical address translation, Addressing modes, Instruction set of 8086 in detail, 8086 Programming examples. Assembly Language Programming, Assembler, Linker, Debugger, Directives, Procedures (Near & Far), Macros, Loop constructs
2	**Memory and I/O Interfacing:** Memory/I/O Read/write cycle timing diagrams, Memory and I/O space of 8086. Memory organization: even and odd banks. Address decoding for memory/IO interfacing. Using NAND gate and Decoder. Programmable Peripheral Interface 8255: Features, Block Diagram, Control & status registers, Operating modes & Interfacing Concept of ADC and DAC
3	**Study of Peripherals:** Serial Communication- Synchronous &Asynchronous, Universal Synchronous Asynchronous Receiver Transmitter USART 8251: Features, Block Diagram, Control & status registers, Operating modes, Interfacing & Programming. Programmable Interrupt Controller 8259: Features, Block Diagram, ICWs, Operating modes & Interfacing
4	**Math-Controller & Study of Motherboard:** 8087(NDP) - Features, Block Diagram, Control & status registers, typical Instruction Set & Programming. Study of IBM PC Motherboard, functional description of Blocks (Overview of 8237, 8254, 8279).

Laboratory

Sr. No.	Contents	Lab Hours
1	**Character String Display** Accept a character string from user and display the same.	02
2	**Hex Number Display** Accept a 4 digit HEX number from user and display the same on screen.	02
3	**Array Addition** Write X86/64 Assembly language program (ALP) to add array of N hexadecimal numbers stored in the memory. Accept input from the user.	02
4	**String Operations** Write X86/64 ALP for the following operations on the string entered by the user. Make your program user friendly by providing MENU like: (a) Enter the string b) Calculate length of string c) Check palindrome d) Exit. Display appropriate messages to prompt the user while accepting the input and displaying the result. Do not use string instructions.	04
5	**String Operations using FAR Procedure** Write 8086 ALP to perform string manipulation. The strings to be accepted from the user is to be stored in data segment of program_1 and write FAR PROCEDURE in code segment program_2 for following operations on the string: (a) Concatenation of two strings Use GLOBAL and EXTERN directive. Create .OBJ files of both the modules and link them to create an EXE file.	04
6	**NDP** Write 8087 ALP to obtain Mean for a given set of data elements defined in data segment. Also display the result.	02
7	**8086 DAC interfacing using 8255** Write 8086 ALP to interface DAC through 8255 and generate following waveforms on oscilloscope, (i) Square wave - Variable Duty Cycle and Frequency. (ii) Ramp wave - Variable direction (iii) Trapezoidal wave (iv) Stair case wave	04

8	**8086 interfacing with 8251** Perform an experiment to establish communication between two 8251 systems A and B. Program 8251 system A in asynchronous transmitter mode and 8251 system B in asynchronous receiver mode. Write an ALP to transmit the data from system A and receive the data at system B. The requirements are as follows: **Transmission:** Message is stored as ASCII characters in the memory. Message specifies the number of characters to be transmitted as the first byte. **Reception:** Message is retrieved and stored in the memory. Successful reception should be indicated.	04

COURSE STRUCTURE

Course Code	**CS223**			
Course Category	**Professional Core**			
Course Title	**Data and Mobile Communication**			
Teaching Scheme and Credits	**L**	**T**	**Laboratory**	**Credits**
Weekly load hrs	**3**	**-**	**2**	**2 + 1 = 3**

Pre-requisites:

- CSIT I & II
- Computer Organization

Course Objectives:

1. To comprehend fundamentals of data communication
2. To study various analog and digital transmission techniques
3. To understand the flow and error control methods
4. To learn basics of cellular telephony .

Course Outcomes:

After completion of the course the students will be able to :-

1. Explain basic concepts of communication system.
2. Identify characterizing features of transmission techniques
3. Apply an appropriate techniques to solve the flow & error control problems
4. Understand wireless communication and cellular architecture

Course Contents:

1. Basics of Communication System
2. Digital Transmission
3. Data Flow and Error Control
4. Fundamentals of Mobile Communication

Laboratory Exercises / Practical:

1. LAN setup
2. Modulation Technique
3. PCM modulation
4. Error detection and correction
5. Sliding window protocol
6. Line coding techniques
7. PC to PC communication

Learning Resources:

Text Books:

1. Behrouz A. Forouzan, " Data Communications and Networking", 5th Edition, McGraw-Hill Publications.
2. Jochen H. Schiller, "Mobile Communications", 2nd Edition, Pearson Education.

Reference Books:

1. William Stallings, "Data and Computer Communications", 6th Edition, Prentice Hall of India Pvt.
2. William C. Y. Lee, " Mobile Cellular Telecommunications: Analog and Digital Systems", 2nd Edition, McGraw- Hill Publications.

Supplementary Readings:

1. Prakash C. Gupta, "Data Communication and Computer Network", PHI, 4th Edition.

Web Resources:

E-books

Web links:

http://www.linktionary.com/re/datacomm.html

MOOCs:

http://nptel.ac.in/courses/106105082/
https://www.coursera.org/learn/data-communication-network-services

Pedagogy:

- Power Point Presentation
- White-board and Pen

Assessment Scheme:

Class Continuous Assessment (CCA)- 50 Marks

Assignments	Test	Tutorial	MCQ	Attendance
20%	30%	-	30%	20%

Laboratory Continuous Assessment (LCA)-50 Marks

Practical	Oral based on practical	Site Visit	Mini Project	Problem based Learning	Any other (Mock 1 and Mock 2)
60%	-	-	-	-	40%

Term End Examination: 50 Marks (100% weightage)

Syllabus :

Unit No.	Contents
1	**Basics of Communication System:** Introduction to Data Communication, Analog and Digital Signal, Baseband Transmission, Transmission Impairment, Data rate limits, Data Transmission (parallel and serial: synchronous and asynchronous transmission), Analog Transmission: Analog-to-Analog conversion, Digital-to-Analog conversion.
2	**Digital Transmission** Digital Transmission: Digital-to-Digital conversion (line coding, block coding, scrambling), Analog-to-Digital conversion (Pulse Code Modulation, Delta Modulation), Multiplexing techniques: Frequency Division Multiplexing, Time Division Multiplexing, Network Model, Networking Devices
3	**Data Flow and Error Control** Error Detection and Correction: Types of Errors, Redundancy, Hamming Distance, Hamming Code, Cyclic Redundancy Check, Checksum, Framing: Fixed Size and Variable Size, Flow and Error Control, protocols: Noiseless and Noisy Channels.
4	**Fundamentals of Mobile Communication** Introduction to Wireless communication, Spread Spectrum: Frequency Hopping Spread Spectrum and Direct Sequence Spread Spectrum, FDMA, TDMA, CDMA, Cellular Telephony: Frequency-Reuse Principle, Handoff, Roaming, Mobile Generations, Architecture of GSM.

Sr. No.	Contents	Lab Hours
1	**LAN setup** Establish wired/wireless network using: UTP Cable,	02

	Crimping Tool, Connectors etc.	
2	**Modulation Technique** Analyze and test Amplitude / Frequency Modulation Technique.	02
3	**PCM modulation** Analyze and test PCM modulation technique.	02
4	**Error detection and correction** Write a program for error detection and correction codes using Hamming Codes/CRC.	04
5	**Sliding window protocol** Write a program to implement sliding window protocol.	04
6	**Line coding techniques** Analyze and test different line coding techniques.	02
7	**PC to PC communication** Study PC to PC communication using RS232 or USB.	02

COURSE STRUCTURE

Course Code	**CS231**			
Course Category	**Professional Core**			
Course Title	**Data Structure-II**			
Teaching Scheme and Credits	**L**	**T**	**Laboratory**	**Credits**
Weekly load hrs	**3**	**--**	**2**	**2+1=3**

Pre-requisites:

- Data Structure-I

Course Objectives:

1. To study algorithms related to trees and graphs.
2. To understand the concept of symbol table.
3. To realize appropriate data structures to solve problems in various domains.

Course Outcomes:

After completion of the course the students will be able to :-

1. Select appropriate data structures in problem solving.
2. Implement various algorithms related to trees and graphs.
3. Demonstrate the use of trees for symbol table implementation.

Course Contents:

1. Tree
2. Graph
3. Hashing
4. Heap
5. Symbol Table
6. APIs for Data Structure

Laboratory Exercises / Practicals:

1. Creation of binary tree and traversal
2. Binary search tree operations
3. Adjacency list representation and DFS & BFS traversals
4. Prim's algorithm for Minimum spanning tree
5. Linear probing with and without replacement
6. AVL tree implementation

Learning Resources:

Text Books:

1. Horowitz, Sahani, Dinesh Mehta, "Fundamentals of Data Structures in C++", Galgotia Publisher, ISBN: 8175152788, 9788175152786.
2. Peter Brass, "Advanced Data Structures", Cambridge University Press, ISBN: 978-1-107-43982.

Reference Books:

1. Sartaj Sahani, "Data Structures, Algorithms and Applications in C++", Second Edition, University Press, ISBN: 81-7371522 X.
2. Augenstein ,Tenenbaum & Langsam,"Data Structure Using C & C++",PHI Publication.

Supplementary Reading:

1. Yashwant Kanitkar," Data Structures through C++", BPB Publication.

Web Resources:

https://www.khanacademy.org/computing/computer-science/algorithms
https://www.hackerrank.com/contests/basic-ds-quiz-1/

Web links:

https://www.tutorialspoint.com/data_structures_algorithms/

MOOCs:

http://nptel.ac.in/courses/106102064/1
https://nptel.ac.in/courses/106103069/

Pedagogy:

- Chalk and Board
- Power Point Presentations
- Two Teacher Method
- Video Lectures
- Discussion Forum
- Flipped Classroom

Assessment Scheme:

Class Continuous Assessment (CCA)-50 Marks

Assignments	Test	Tutorial	MCQ	Attendance
20%	30%	-	30%	20%

Laboratory Continuous Assessment (LCA)-50 Marks

Practical	Oral based on practical	Site Visit	Mini Project	Problem based Learning	Any other (Mock 1 and Mock2)
60%	-	-	-	-	40%

Term End Examination: 50 Marks (100% weightage)

Syllabus:

Unit No.	Contents
1	**Tree-** Basic Terminology, Binary Tree- Properties, Converting Tree to Binary Tree, Representation using Sequential and Linked organization, Binary tree creation and Traversals, Operations on binary tree. Binary Search Tree (BST) and its operations, Threaded binary tree- Creation and Traversal of In-order Threaded Binary tree. **Case Study**- Expression tree
2	**Graph-** Basic Terminology, Storage representation**:** Adjacency matrix, Adjacency list, Creation of Graph and Traversals, Minimum spanning Tree- Prim's and Kruskal Algorithms, Topological sorting
3	**Hashing-** Introduction to hashing**,** Hash functions, Collision resolution strategies- Open Addressing and Chaining, Hash Table Overflow. **Heap**- Heap as a priority queue, Heap sort.
4	**Symbol Table-**Introduction to Symbol Tables, Static tree table- Optimal Binary Search Tree (OBST), Dynamic tree table-AVL tree, Multi way search tree- B-Tree. **APIs for Data Structure**- Standard Template Library(STL) for data structures.

Laboratory:-

Sr. No.	Contents	Lab Hours
1	Implement binary tree using C++ and perform following operations: Creation of binary tree and traversal (recursive and non- recursive)	04
2	Implement dictionary using binary search tree where dictionary stores keywords & its meanings. Perform following operations: i. Insert a keyword ii. Delete a keyword iii. Create mirror image and display level wise iv. Copy	04
3	Consider a friend's network on Facebook social web site. Model it as a graph to represent each node as a user and a link to represent the friend relationship between them using adjacency list representation and perform DFS & BFS traversals.	04
4	A business house has several offices in different countries; they want to lease phone lines to connect them with each other and the phone company charges different rent to connect different pairs of cities. Business house wants to connect all its offices with a minimum total cost. Solve the problem using Prim's algorithm.	02
5	Store data of students with roll no, name and grade. Implement linear probing with and without replacement.	02
6	Implement AVL tree for mnemonics of assembly language and display using inorder traversal.	04

Dr. Vishwanath Karad
MIT WORLD PEACE UNIVERSITY | PUNE
TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

COURSE STRUCTURE

Course Code	**CS232**			
Course Category	**Professional Core**			
Course Title	**Microprocessors and Microcontrollers**			
Teaching Scheme and Credits	**L**	**T**	**Laboratory**	**Credits**
Weekly load hrs	**3**	**-**	**2**	**2 + 1 = 3**

Pre-requisites:

- Computer Organization
- Microprocessor and Interfacing Techniques

Course Objectives:

1. To learn the architecture and programming of Pentium Microprocessor.
2. To learn the architecture and programming of 8051 Microcontroller.
3. To understand the architectural design of advanced microprocessors and microcontrollers.

Course Outcomes:

After completion of the course the students will be able to :-

1. Understand the advanced architectural features of various processors.
2. Analyze use of microprocessor or microcontroller depending on requirements of application.
3. Write assembly language programs for 8051 microcontroller based systems.

Course Contents:

1. Pentium Architecture
2. Memory Management in Protected Mode
3. Multitasking and Interrupts, Exceptions and I/O
4. 8051 Microcontroller

Laboratory Exercises / Practicals:

1. Counting of positive and negative numbers.
2. Sorting of numbers from an array.
3. Hex to BCD conversion.
4. Simulation of 'cat' command.
5. Simulation of 'cp' command.
6. Display contents of system registers in Pentium.
7. Timer programming in 8051.
8. Serial port programming in 8051.

Learning Resources:

Text Books:

1. James Antonakos , "The Pentium Microprocessor" , 2004, Pearson Education ISBN – 81-7808-545-3.
2. K.J.Ayala " The 8051 Microcontroller " ISBN 9788131511053.

Reference Books:

1. Intel architecture software developer's manual volume 3.
2. Intel architecture software developer's manual volume 1.
3. Intel 8051 datasheet.

Supplementary Reading: -

Web Resources:

E-books

Web links:

nptel.ac.in/courses/.../IIT.../Course_home2_5.htm
nptel.ac.in/courses/.../IIT.../Course_home4_35.htm

MOOCs:

nptel.ac.in/courses/.../IIT.../Course_home2_5.htm
nptel.ac.in/courses/.../IIT.../Course_home4_35.htm

Pedagogy:

- White Board
- Power Point Presentations
- Two Teachers method

Assessment Scheme:

Class Continuous Assessment (CCA)- 50 Marks

Assignments	Test	Tutorial	MCQ	Attendance
20%	30%	--	30%	20%

Laboratory Continuous Assessment (LCA)-50 Marks

Practical	Oral based on practical	Site Visit	Mini Project	Problem based Learning	Mock 1 and Mock 2
60%	--	--	--	--	40%

Term End Examination: 50 Marks (100% weightage)

Syllabus :

Unit No.	Contents
1	**Pentium Architecture** : Pentium features, Operating Modes, Pentium super-scalar architecture - Pipelining, Branch prediction, Instruction and Data caches. Pentium programmer's model, Register set, Addressing modes and Instruction set. The Floating point Unit: features, pipeline stages & data types.
2	**Memory Management in Protected Mode:** Segmentation unit : Introduction, support registers, related instructions, segment descriptors, logical to linear address translations, protection by segmentation, privilege-levels, rules of inter-privilege level transfer for data and code segments . Paging Unit: support registers, related data structures ,linear to physical address translation ,TLB ,page level protection.
3	**Multitasking and Interrupts, Exceptions and I/O :** Task Management -support registers, related data structures, Task switching. Interrupt and Exception Handling : IDT and Gate descriptors. I/O handling and I/O instructions in Pentium.
4	**8051 Microcontroller:** Features, Micro-controller MCS-51 family architecture. Programmers model-register set, register bank, SFRs, addressing mode, instruction set, Memory organization :on-chip and external memory components. Interrupt structure, Timers and their programming, Serial port and programming.

Sr. No.	Contents	Lab Hours
1	Write an ALP to find positive and negative numbers from an array of 32 bit numbers.	02
2	Write an ALP to sort 8 bit numbers in ascending/descending order.	02
3	Write an ALP to convert 4 digit HEX number to 5 digit BCD number.	02
4	Write an ALP to simulate 'cat' command in linux.	02
5	Write an ALP to simulate 'cp' command in linux.	02
6	Write an ALP to display the contents of GDTR, IDTR, LDTR, TR and MSW .	02

7	Write an 8051 ALP for rate generation using Timer0/Timer1 by using a. Polling method b. ISR method	02
8	Write an 8051 ALP for serial port programming to transfer block of data using a. Polling method b. ISR method	02

COURSE STRUCTURE

Course Code	**CS233**			
Course Category	**Professional Core**			
Course Title	**Software Engineering and Project Management**			
Teaching Scheme and Credits	**L**	**T**	**Laboratory**	**Credits**
Weekly load hrs	**3**	**-**	**-**	**2**

Pre-requisites:

- Principles of Programming Languages and Functional Programming
- Object Oriented Programming

Course Objectives:

1. To identify the systematic way of Software Development and Software Lifecycle Process model.
2. To identify the principles of Agile Software Development and Practices.
3. To explain the Project Management principles and Project Metrics

Course Outcomes:

After completion of this course students will be able to:

1. Analyze process models and its appropriate selection for Development of Software Projects.
2. Define scope of Project, design Software Requirement Specifications (SRS) and plan a project.
3. Make use of the principles and processes of Agile Methodology.
4. Choose modern tools for Software Development and Project Management.

Course Contents:

1. Software Process
2. Requirement Engineering
3. Agile Management and Practices
4. Software Project Management

Learning Resources:

Text Books:-

1. Roger S Pressman, "Software Engineering: A Practitioner's Approach", McGraw-Hill, ISBN:
   0073375977, Seventh Edition, 2014.
2. Joseph Phillips, "IT Project Management –On Track From Start to Finish", Tata McGraw-Hill.2009.
3. The Art of Agile Development James Shore, Shane Warden "O'Reilly Media, Inc." 2008.

Reference Books:-

1. Pankaj Jalote, "Software Engineering : A Precise Approach", Wiley India.2010.
2. P.C. Tripathi, P.N. Reddy, "Principles of Management", Tata McGraw Hill Education Private Limited. 2012.

Supplementary Reading:

1. Ian Sommerville, ― Software Engineering‖, Addison and Wesley. 9th Ed., 2011.

Web Resources:

http://highered.mheducation.com/sites/0072853182/student_view0/index.html
https://poetiosity.files.wordpress.com/2011/04/art_of_agile_development.pdf

Weblinks:

https://www.sei.cmu.edu/training/p35.cfm
http://www.mhhe.com/engcs/compsci/pressman/

MOOCs:

http://nptel.ac.in/courses/106101061/1
http://nptel.ac.in/downloads/106105087/

Pedagogy:

- Chalk and Board
- Power Point Presentation
- Two Teacher Method
- Video Lectures
- Group Discussion

Assessment Scheme:

Class Continuous Assessment (CCA): 50 Marks

Assignments	Test	Tutorial	MCQ	Attendance
20%	30%	-	30%	20%

Laboratory Continuous Assessment (LCA):50 Marks

Practical	Oral based on practical	Site Visit	Mini Project	Problem based Learning	Any other (Mock 1 and Mock2)
60%	-	-	-	-	40%

Term End Examination: 50 Marks (100% weightage)

Syllabus:

Unit No.	Contents
1	**Software Process** Introduction to Software Engineering, Software Engineering Practice, Software Myths, Software Process, Perspective and Specialized Process Models. Case Studies.
2	**Requirement Engineering** Software Requirements, User requirements, System requirements, Software Requirements Specification, Requirement Engineering Process, Requirements validation techniques, Requirement Traceability Matrix, Change Management, Software Design: Abstraction, Modularity, Cohesion & Coupling, Scenario based modeling, Introduction to Unified Modeling Language (UML) and Data Flow Diagram (DFDs). Case Studies. .
3	**Agile Management and Practices** Agile manifesto, The Fundamentals of Agile Software Development, Aspects of Agile Approaches, Practices and Processes, Techniques in Agile Projects, Extreme Programming, Other process models of Agile Development and Tools.

	Tools in Agile Projects-Case Studies.
4	**Software Project Management** Project Management Principles , Process and Project Metrics., Function Point analysis, LOC , Make/Buy Decision, COCOMO II -Project Planning, SWOT analysis, Functions of manager, Team building & development, Request for Proposal (RFP), Risk Management – Identification, Projection, CPM/PERT, RMMM - Scheduling and Tracking , EVA . Case Studies and Examples.

COURSE STRUCTURE

Course Code	**CS234**			
Course Category	**Professional Core**			
Course Title	**Operating Systems**			
Teaching Scheme and Credits	**L**	**T**	**Laboratory**	**Credits**
Weekly load hrs	**3**	**-**	**2**	**2 + 1 = 3**

Pre-requisites:

- Fundamentals of Computers
- Data Structures
- Computer Organization

Course Objectives:

1. To discuss a generic overview of operating systems.
2. To describe the concepts of process management.
3. To state the concepts related to synchronization of processes.
4. To explain the concepts of Memory Management and I/O management.

Course Outcomes:

After completion of the course the students will be able to :-

1. Comprehend key mechanisms of the Operating System functions.
2. Demonstrate processor scheduling algorithms.
3. Explain solutions to process synchronization problems.
4. Assess memory management issues.

Course Contents:

1. Overview of Operating System.
2. Process Management.
3. Concurrency Control.
4. Memory Management, I/O and File Management.

Laboratory Exercises / Practical:

1. Shell Programming.
2. Process Control.
3. CPU Scheduling .
4. Bankers algorithm for deadlock avoidance.
5. POSIX threads (pthread) to perform arithmetic operations.
6. Page Replacement Algorithms.

Learning Resources:

Text Books:

1. William Stallings, Operating System: Internals and Design Principles, Prentice Hall, ISBN-10: 0-13-380591-3, ISBN-13: 978-0-13-380591-8, 8th Edition.
2. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, Operating System Concepts, WILEY,ISBN 978-1-118-06333-0, 9th Edition.

Reference Books:

1. Harvey M. Deitel, Operating Systems, Prentice Hall, ISBN-10: 0131828274, ISBN-13: 978-0131828278.
2. W. Richard Stevens, Stephen A. Rago, Advanced Programming in the UNIX Environment, Addison- Wesley Professional, ISBN: 9780321637734,3rd Edition.
3. Sumitabha Das, Unix concepts and applications ,McGraw Hill,ISBN-13-978-0-07063546-3,4th Edition.

Supplementary Reading:

1. Andrew Tanenbaum, Modern Operating Systems,Peasrson,4th Edition.

Web Resources:

E-books
http://engineeringppt.blogspot.in/2009/07/operating-system-concepts-8th-edition.html

Web links:

https://www.google.co.in/search?q=advanced+programming+in+unix+environment&ie=utf-8&oe=utf-8&client=firefox-b&gfe_rd=cr&dcr=0&ei=5khOWtHyCK_T8geE65jQAQ
http://williamstallings.com/OperatingSystems/

MOOCs:

https://in.udacity.com/course/introduction-to-operating-systems--ud923
http://nptel.ac.in/courses/106108101/

Pedagogy:

- Power Point Presentation
- White-board / Pen
- Two Teacher method

Assessment Scheme:

Class Continuous Assessment (CCA)- 50 Marks

Assignments	Test	Tutorial	MCQ	Attendance
20%	30%	-	30%	20%

Laboratory Continuous Assessment (LCA)-50 Marks

Practical	Oral based on practical	Site Visit	Mini Project	Problem based Learning	Any other (Mock 1 and Mock2)
60%	-	-	-	-	40%

Term End Examination: 50 Marks (100% weightage)

Syllabus :

Unit No.	Contents
1	**Overview of Operating System** Operating System objectives and its evolution. Operating System structure: Layered, Monolithic, Microkernel Operating Systems. BASH shell scripting.
2	**Process Management** **Process:** Concept of a Process, Process States, Process Control-creation, new program execution, termination. **Threads:** Processes and Threads, Concept of Multithreading, Types of Threads, Thread programming Using Pthreads. **Scheduling:** Types of Scheduling, Scheduling Algorithms: FCFS, SJF, Priority, Round Robin.
3	**Concurrency Control** **Process Synchronization:** Principles of Concurrency, Requirements for Mutual Exclusion, Mutual Exclusion: Hardware Support, Operating System Support (Semaphores and Mutex). **Classical synchronization problems:** Readers/Writers Problem, Producer and Consumer problem. **Deadlock:** Principles of Deadlock, Deadlock Modeling, Deadlock Prevention, Deadlock Avoidance, Deadlock detection and recovery.
4	**Memory Management, I/O and File Management** **Memory Management:** Memory Management Requirements, Memory Partitioning: Fixed Partitioning, Dynamic Partitioning, paging, segmentation, virtual memory. **File Management:** Overview, File Organization and Access, File Directories, File Sharing, Record Blocking.

	I/O Management: I/O Devices, Organization of the I/O Functions, I/O Buffering, Disk Scheduling.
Sr. No.	**Laboratory Contents**
1	**Shell Programming** Write a shell script to:- • Display odd-even numbers user specified limit. • Convert a string from upper case to lower case or vice versa
2	**Process Control** Write a program using fork to create a child process. The parent process should sort elements in ascending order and child process should sort elements in descending order.
3	**CPU Scheduling** Write a menu driven program to simulate the following CPU scheduling algorithms :- • First Come First Serve (FCFS). • Round Robin (RR).
4	**Deadlocks** Write a program to simulate Bankers algorithm for deadlock avoidance.
5	**Multi-Threading** Write a program using POSIX threads (pthread) to perform arithmetic operations
6	**Page Replacement Algorithms** Write a menu driven program to simulate the following page replacement algorithms. • First In First Out (FIFO). • Least Recently Used (LRU).

COURSE STRUCTURE

Course Code	**IC**			
Course Category	**Humanities and Social Science**			
Course Title	**Indian Constitution**			
Teaching Scheme and Credits	**L**	**T**	**Laboratory**	**Credits**
Weekly load hrs	**2**	**--**	**--**	

Pre-requisites: Knowledge of Civics

Course Objectives:

1. To provide basic information about Indian constitution.
2. To identify individual role and ethical responsibility towards society.

Course Outcomes:

After study of the course, the students are able to

1. Have general knowledge and legal literacy and thereby to take up competitive examinations
2. Understand state and central policies, fundamental duties • Understand Electoral Process, special provisions
3. Understand powers and functions of Municipalities, Panchayats and Co-operative Societies, and
4. Understand Engineering ethics and responsibilities of Engineers.
5. Have an awareness about basic human rights in India

Course Contents:

1. Introduction to the Constitution of India, The Making of the Constitution and Salient features of the Constitution.
2. Preamble to the Indian Constitution Fundamental Rights & its limitations.
3. Directive Principles of State Policy & Relevance of Directive Principles, State Policy, Fundamental Duties.
4. Union Executives – President, Prime Minister Parliament Supreme Court of India.
5. State Executives – Governor Chief Minister, State Legislature High Court of State.
6. Electoral Process in India, Amendment Procedures, 42 nd, 44th, 74th, 76th, 86th &91st Amendments.
7. Special Provision for SC & ST Special Provision for Women, Children & Backward Classes Emergency Provisions. Human Rights –Meaning and Definitions, Legislation Specific Themes in Human Rights- Working of National Human Rights Commission in India
8. Powers and functions of Municipalities, Panchyats and Co – Operative Societies.

Learning Resources:

Reference Books: Durga Das Basu: "Introduction to the Constitution on India", (Students Edn.) Prentice –Hall EEE, 19th / 20th Edn., 201 2.

1. Charles E. Haries, Michael S Pritchard and Michael J. Robins "Engineering Ethics" Thompson Asia, 2003-08-05.

Web Resources:

Web-links:

MOOCs:

Pedagogy: Power Point Presentation, Quizzing, Interactive Discussions, site visits

Assessment Scheme:

Class Continuous Assessment (CCA) 50 Marks

Assignments	Test	Presentations	MCQ	Any other
60%	--	40%	--	--

Syllabus :

Sr. No.	Lecture Plan
1	Introduction to the Constitution of India, The Making of the Constitution and Salient features of the Constitution.Preamble to the Indian Constitution Fundamental Rights & its limitations.
2	Directive Principles of State Policy & Relevance of Directive Principles State Policy Fundamental Duties. Union Executives – President, Prime Minister Parliament Supreme Court of India.
3	State Executives – Governor Chief Minister, State Legislature High Court of State.Electoral Process in India, Amendment Procedures, 42 nd, 44th, 74th, 76th, 86th &91st Amendments.
4	Special Provision for SC & ST Special Provision for Women, Children & Backward Classes Emergency Provisions. Human Rights –Meaning and Definitions, Legislation Specific Themes in Human Rights- Working of National Human Rights Commission in India Powers and functions of Municipalities, Panchyats and Co – Operative Societies.