Proposed Structure for Third Year Engineering (CSE) from Academic Year 2018- 19 (CBCS Pattern)

Semester- I

Sr. No	Code	Subject	Contact Period (Hrs.)			Credits	Continuous Evaluation in terms of Marks						
			L	T	P		Class Test I	Class Test II	TA	ESE	TW	PR	Total
1	CS3001	Design & Analysis of Algorithm	3	1	-	4	15	15	10	60	-		100
2	CS3002	Operating System	3		-	3	15	15	10	60	-		100
3	CS3003	Theory of Computation	3	1	-	4	15	15	10	60	-		100
4	CS3004	Software Engineering	3	1	-	4	15	15	10	60	-		100
5	CS3005	Data Mining & Warehousing	3	1	-	4	15	15	10	60	-		100
6	CS3006	Lab Design & Analysis of Algorithm	-	-	2	1		-	-	-	25	25	50
7	CS3007	Lab Operating System	-	-	2	1		-	-	-	25	25	50
8	CS3008	Lab Data Mining & Warehousing			2	1					25	25	50
9	CS3009	Mini Project			4	2					25	25	50
		Total	**15**	**4**	**10**	**24**	**75**	**75**	**50**	**300**	**100**	**100**	**700**

Semester- II

Sr.No	Code	Subject	Contact Period (Hrs.)			Credits	Continuous Evaluation in terms of Marks						
			L	T	P		Class Test I	Class test II	TA	ESE	TW	PR	Total
1	CS3010	Computer Network	3	-	-	3	15	15	10	60	-	-	100
2	CS3011	System Programming & Compiler Construction	3	1	-	4	15	15	10	60	-	-	100
3	CS3012	Advanced Algorithm	3	1	-	3	15	15	10	60	-	-	100
4	CS3013	Data Science	3	1	-	4	15	15	10	60	-	-	100
5	CS30XX	Open Elective-I	3	-		3	15	15	10	60	-	-	100
6	CS3014	Lab Advanced Development Tool	-	-	2	1	-	-	-	-	25	25	50
7	CS3015	Lab Computer Network	-	-	2	1	-	-	-	-	25	25	50
8	CS3016	Lab Advanced Algorithm	-	-	2	1	-	-	-	-	25	25	50
9	CS3017	Lab Data Science	-	-	2	1	-	-	-	-	25	25	50
10	CS3018	Seminar	-	-	2	1	-	-	-	-	25	-	25
		Total	**15**	**3**	**10**	**22**	**75**	**75**	**50**	**300**	**125**	**100**	**725**
		Grand Total	**30**	**7**	**20**	**46**	**150**	**150**	**100**	**600**	**225**	**200**	**1425**

L = Lecturer, **T** = Tutorial, **P** = Practical, **TA** = Teacher Assessment, **ESE** = End Semester Examination

CS3001: Design and Analysis of Algorithm

Teaching Scheme		**Evaluation Scheme**	
Lectures	3 Hrs/Week	Test 1	15 Marks
		Test 2	15 Marks
Tutorials	1 Hrs/Week	Teacher Assessment	10 Marks
Total Credits	4	End-Semester Examination	60 Marks

Total Hours required for this course: 60 Hours.

Prerequisite: CS1001: Basics of Computer & Information Technology
CS2003: Discrete Mathematical Structure
CS2008: Data Structures

Course description: This Course describes the techniques of design and analysis of algorithms. Topics like Divide and Conquer, Greedy and Dynamic programming, Backtracking will be covered. Students will also learn to analyze the performance of algorithms

Course Educational Objectives:

- Demonstrate knowledge of how to measure the complexity of an algorithm
- Understand best-case, worst-case, and average complexities
- Know the basic algorithmic design strategies, including recursion, divide-and-conquer, greedy method
- Learn dynamic programming and backtracking.
- Use data structure in providing efficient algorithms solutions.

Course Outcomes Expected:

After completion of this course students will be able to:

CO1: Describe asymptotic notation, its properties and use in measuring algorithm behavior.
CO2: Apply mathematical principles to solve various problems
CO3: Analyze and apply the complexities of various algorithms and select the best
CO4: Know the different strategies that are known to be useful in finding efficient algorithms to solve problems and to be able to apply them.
CO5: Use appropriate data structure and algorithms to solve a particular problem

Detailed Syllabus

UNIT-1 DIVIDE-AND-CONQUER

What is an algorithm, Performance Analysis- Space complexity, Time Complexity, Asymptotic Notation, and Divide-and-Conquer- Introduction, Binary Search- Iterative and Recursive, finding the Maximum and Minimum, Merge Sort, Quick Sort, Heap Sort.

UNIT-2 GREEDY METHOD

Introduction , 0/1 Knapsack Problem, Job scheduling, Huffman codes, Minimum cost spanning trees- Prim's Algorithm, Kruskal's Algorithm, Optimal Merge Patterns.

UNIT-3 DYNAMIC PROGRAMMING

Multistage Graphs, All pairs shortest path, single source shortest path, Optimal Binary Search tree, Traveling Sales man problem, Flow shop Scheduling.

UNIT-4 BACK TRACKING

Introduction, The 8 queens problem, Sum of Subset, Graph coloring, Hamiltonian cycles ,Branch and Bound

UNIT-5 BASIC TRAVERSAL AND SEARCH TECHNIQUES

Techniques for binary trees, Techniques for graphs- Breadth First Search and traversal, Depth First Search and traversal, Connected components and Spanning Trees.

TEXT BOOKS

1. Horowitz, Sahani, Rajasekaran, Fundamental of Computer Algorithm , Galgotia Publication
2. Coreman, Leiserson, Rivest, Stein, Introduction to Algorithms, PHI.

REFERENCE BOOKS

1. AhoUlman, Hopecroft, Design and Analysis of Algorithms, Addison Wesley

Special Instructions if any: Nil

Designed by

- Mr. Sudhir G. Shikalpure

CS3002: Operating Systems

Teaching Scheme		**Evaluation Scheme**	
Lectures	3 Hrs/Week	Test 1	15 Marks
		Test 2	15 Marks
		Teacher Assessment	10 Marks
Total Credits	3	End-Semester Examination	60 Marks

Total Hours required for this course: 60 Hours.

Prerequisites: CS2009:Microprocessor and Interfacing
CS1001: Basics of Computer & Information technology
CS2008: Data Structures

Course Description: Topics will include what an operating system does ,management of the CPU, memory, processes and devices with exposure to android operating sytem.

Course Educational Objectives:

- Define and list the functions of an operating system.
- list resources involved in process creation and management.
- Explain the use of paging and segmentation
- Explain the function and structure of the I/O system
- Describe path names and directory structure visible to end users

Course Outcomes: After completion of this course students will be able to:

CO1: Differentiate between multiprocessing, multiprogramming, and multitasking.
CO2: Differentiate between programs, processes and threads.
CO3: Apply segmentation and paging techniques.
CO4: Compare file naming in Linux and Windows.
CO5:Awareness of Android Operating System

Detailed Syllabus:

UNIT 1 Fundamentals of Operating System:-
OS services and Components, Multitasking , Multiprogramming, Multiprocessing Time Sharing, Buffering, Spooling, Distributed OS

UNIT 2 Process and Thread Management
Concept of process and threads ,Process states ,Process management ,Context switching ,Interaction between processes and OS ,Multithreading, Example OS : Linux

UNIT 3 Memory Management
Memory partitioning, Swapping, Paging ,Segmentation ,Virtual memory Overlays ,Demand paging ,Performance of Demand paging ,Virtual memory concepts ,Page replacement algorithms ,Allocation algorithms ,Example OS : Linux

UNIT 4 I/O Systems
Secondary-Storage Structure, Disk structure ,Disk scheduling ,Disk management ,Swap-space management ,Disk reliability ,Stable storage implementation ,Introduction to clock ,Clock hardware ,Clock software

UNIT 5 File systems

File concept ,File support ,Access methods ,Allocation methods ,Directory systems ,File protection ,Free space management
Example OS : Linux Case study : Android OS

TEXT BOOKS:

1. Abraham Silberschatz, Peter B. Galvin, Greg Gagne, Operating System Concepts. Seventh edition. Addison-Wesley
2. Andrew Tanenbaum, Modern Operating Systems, Prentice Hall.

REFERENCE BOOK:

1. Operating Systems (5th Ed) – Internals and Design Principles By WilliamStallings, Prentice Hall
2.Operating Systems Achyut S. Godbole Tata Mc Graw Hill

Special Instructions if any: Nil

Designed by

- Mr. Vivek Kshirsagar
- Mrs. Meghana B. Nagori
- Mrs. Vijayshri A. Injamuri

CS 3003: Theory of Computation

Teaching Scheme		**Evaluation Scheme**	
Lectures	3 Hrs/Week	Test 1	15 Marks
		Test 2	15 Marks
Tutorials	1 Hrs/Week	Teacher Assessment	10 Marks
Total Credits	4	End-Semester Examination	60 Marks

Total Hours required for this course: 60 Hours.

Prerequisite: CS2003: Discrete Mathematical Structures

Course Description: This course provides a set of abstract structures that are useful for solving certain classes of problems. It describes properties and design methods and corresponding languages of Finite automata, pushdown automata and Turing machine.

Course Educational Objectives:

- Demonstrate different language processing abstract machines.
- Explain relationship between different languages and automata
- Illustrate string membership problems
- Design automata and language under specific criteria.

Course Outcomes Expected:

After completion of this course students will be able to:

CO1: Discuss properties of different grammars and languages (K1)

CO2: Solve problems related to string membership to an automata and respective Language. (K2)

CO3: Create grammar for specific language. (K3)

CO4: Identify language accepted by particular automata. (K2)

CO5: Design optimum automata for particular language (K3)

Detailed Syllabus:

UNIT-1 Automata: why study automata theory? Introduction to formal proof, Inductive Proofs, The central concept of automata theory.
Finite Automata: Deterministic Finite automata, Nondeterministic finite automata, An Application: Text Search, Finite automata with Epsilon – Transitions.

UNIT-2 Regular Expressions and Languages: Regular expressions, Finite automata and regular expressions, Applications of regular expressions, Algebraic Laws for Regular Expressions
Properties of Regular Languages: Proving Languages not to be regular, Closure properties of regular Languages, Decision properties of Regular Languages, Equivalence and minimization of Automata.

UNIT-3 Context Free Grammars and Languages: Context Free Grammars, parse Trees, Application of Context Free Grammars, Ambiguity in Grammars and languages.
Push Down Automata: Definition of the Pushdown Automaton, The Languages of a PDA, Equivalence of PDA's and CFG's, Deterministic Pushdown Automaton.

UNIT-4 Properties of Context Free Languages: Normal Forms for Context Free Grammars, The pumping Lemma for context Free Languages, Closure Properties of Context Free Languages, and decision properties of CFL's.

UNIT-5 Introduction to Turing Machine: Problems that computer cannot solve, The Turing Machine, Programming Techniques for Turing machines, Extensions to the basic Turing Machines, Turing machines and Computers, Undecidable Problems about Turing machines. An Introduction to intractable problems.

TEXT BOOKS

1. Hopcroft&Ullaman, Introduction to Automata Theory languages, and Computation 3rd edition. Pearson Education.
2. John C. Martin , Introduction to Languages and theory of computation 2^{nd} edition TMH.

REFERENCE BOOKS

1. K.L.P.Mishra ,N. Chandrasekaran, Theory of Computer Science Automata, Languages and Computation) , 2^{nd} Edition PHI

Special Instructions if any: Nil

Designed by

- Mrs. Pallavi V. Kulkarni
- Mrs. Meghana B. Nagori

CS3004: Software Engineering

Teaching Scheme		**Evaluation Scheme**	
Lectures	3 Hrs/Week	Test 1	15 Marks
		Test 2	15 Marks
Tutorials	1 Hrs/Week	Teacher Assessment	10 Marks
Total Credits	3	End-Semester Examination	60 Marks

Total Hours required for this course: 45 Hours.

Prerequisite: Nil

Course Description: This course presents fundamentals of software engineering and process related activity for development of software product, this subject gives the knowledge of requirement engineering to collect required information appropriately from customer and maintaining the specification document and it also gives knowledge of design of software product. At the end it gives the idea about how to plan the testing for software product.

Course Educational Objectives:

1. Understand the processes involved in SDLC lifecycle.
2. Know the how requirement can be gathered through requirement engineering.
3. Learn the how to design the system and do the data mapping.
4. Recognize the importance of Golden Rules.
5. Learn the basic software testing strategy.

Course Outcomes Expected:

After completion of this course students will be able to:

CO1: Aware of basic computer engineering concept through SDLC life cycle and Models in software engineering.
CO2: Integrate the requirement from customer for software development.
CO3: Apply the design concept to develop the system.
CO4: Apply the "Golden Rules" for user interface level design.
CO5: Formulate test strategy and ethically work to achieve the quality of product.

UNIT-1 Introduction to Software Development process and Management

Introduction to Software Engineering, Software Crisis and Myths, Software Development life cycle and Models: Maturity Model, Process models-waterfall, evolutionary, incremental etc, What is an agile view of process.

UNIT-2 Requirements Engineering

Requirements Engineering and Management. Initiating, Eliciting requirement, developing use cases, building the Analysis Model, Negotiating and Validating requirement.

UNIT-3 An Architectural Design

Software Architecture, Data Design, Architectural Styles and pattern, Architectural Design, Assessing alternative architectural Design Mapping Data flow in to software architecture

UNIT-4 User Interface Level Design and Estimation

The Golden rules, User Interface analysis and Design, Interface Analysis, Interface Design Steps, Design Evaluation, Project Estimation.

UNIT-5 Introduction to Software Testing

Test Strategies for conventional software, Validation testing, System Testing, The art of Debugging, Software testing fundamentals, Software quality, Framework for product Metrics.

TEXT AND REFERENCE BOOKS

1. Software Engineering – Practitioner Approach – Roger S. Pressman
2. Software Engineering by Ian Sommerville ; Pearson Edu
3. Object Oriented Analysis and Design, Grady Booch

Special Instructions if any: Nil

Designed by

- Mr. Vikul J.Pawar
- Mrs. Vrushali A. Chakkarwar
- Mr. Prashant D. Pathak

CS3005: Data Mining & Warehousing

Teaching Scheme		**Evaluation Scheme**	
Lectures	3 Hrs/Week	Test 1	15 Marks
		Test 2	15 Marks
Tutorials	1 Hrs/Week	Teacher Assessment	10 Marks
Total Credits	4	End-Semester Examination	60 Marks

Total Hours required for this course: 60 Hours.

Prerequisites: MA2001**:** Engineering Mathematics-III, CS1001: Basics of Computer & Information Technology. CS2002: Database Management System

Course Educational Objectives:

1. To introduce students to the basic concepts and techniques of Data Mining
2. To develop Skills of using recent data mining software for solving practical problems
3. To gain experience of doing independent study and research.
4. Explain the significance of Exploratory data analysis.

Course Outcomes Expected:

After completion of this course student will be able to

CO1: Apply EDA in a case study
CO2: Apply basic machine learning algorithms.
CO3: Identify approaches used for feature selection
CO4: Create effective visualization of given data
CO5: Work effectively in teams of data science projects.

Detailed Syllabus:

UNIT-1 Introduction to Data Mining

Related Technologies-Machine Learning, DBMS, OLAP and Statistics, Data Mining Goals, Stages of Data Mining Process, Data Warehouse and Multidimensional Data Model, OLAP Operations.

UNIT-2 Data Preprocessing and Data Visualization

Data Cleaning, Data Transformation, Data Reduction, Discretization and Concept hierarchy, Exploratory Data Analysis Tools(Plots, Graphs, Summary Statistics, histograms, heat maps)

UNIT-3 Data Mining algorithms:-Association Rules and Regression Analysis

Item sets, Frequent Patterns, Interestingness measures(support, confidence and lift),Co-relation analysis,

Apriori and Frequent Growth Algorithms, Linear Regression and Models

UNIT-4 Data Mining Algorithms: Classification

Decision Trees, Random Forests, Bayesian Networks, Nearest Neighbour Algorithms.

UNIT-5 Data Mining Algorithms:Clustering and Model Evaluation:

Partitioning Methods:-K-means,k-Medeoids,ExpectationMaximization,HierarchicalMethods:Distance-Basedagglomertive clustering and divisible clustering,Training And Testing(Cross-Validsation),Combining multiple models(bagging,boosting)

TEXT BOOKS:-

1. Jiawei Han and MichelineKamber, "Data Mining Concepts and Techniques", Second Edition, Elsevier, 2007.
2. Alex Berson and Stephen J. Smith, "Data Warehousing, Data Mining & OLAP", Tata McGraw – Hill Edition, Tenth Reprint 2007.

REFERENCE BOOKS:-

1. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, "Introduction To Data Mining", Person Education, 2007.
2. G. K. Gupta, "Introduction to Data Mining with Case Studies", Easter Economy Edition, Prentice Hall of India, 2006.

1) Special Instructions if any: Nil

Designed by

- Mrs. Meghana B. Nagori
- Mrs. Pallavi V. Kulkarni
- Mrs. Vrushali Chakkarwar

CS 3006 : Lab Design & Analysis of Algorithms

Teaching Scheme		**Evaluation Scheme**	
Practical	2 Hrs/Week	Term Work	25 Marks
Credits	1	Practical/Viva-voce	25 Marks

Total Hours required for this practical course: 30 Hours.

Prerequisite: CS1001: Basics of Computer & Information Technology
CS2003: Discrete Mathematical Structure
CS2008: Data Structures

Course Outcomes:

After completion of this course students will be able to:

CO1	Design and implement appropriate data structures for computation
CO2	Demonstrate algorithms using divide and conquer approach
CO3	Solve problems using greedy method.
CO4	Employ dynamic programming techniques.
CO5	Problem solving Using backtracking techniques

The term work shall consist of following practical/assignments/mini-project/visit*

1	Recursive and non-recursive algorithm for specific problem and there complexity measures
2	Implement merge sort using divide and conquer approach.
3	Write a program for finding an element Using Binary Search.
4	Write a Program for Greedy Knapsack problem.
5	Minimal spanning trees using Prime's algorithm.
6	Minimal spanning trees using Kruskal's algorithm
7	Find single source shortest path for multistage graph problem
8	Find all pairs shortest path for multistage graph problem.
9	Huffman code problem.
10	Flow shop scheduling or knapsack's problem or 8 Queen problem

Special Instructions if any: Nil

Designed by

- Mr. Sudhir G. Shikalpure
- Mrs. Meghana B. Nagori
- Mrs. Pallavi Kulkarni

CS3007: Lab Operating System

Teaching Scheme		**Evaluation Scheme**	
Practical	2 Hrs/Week	Team Work	25 Marks
Credits	1	Practical /Viva-voce	25 Marks

Total Hours required for this practical course: 30 Hours.

Prerequisites: Programming Language, Data Structures

Course Outcome:

After completion of this course student will be able to

CO1: Exposure to different OS

CO2: Awareness of concepts of multiprogramming, multithreading and multitasking

CO3: Demonstration of memory management algorithms

CO4: Demonstration of file-handling concepts by implementing suitable algorithms.

CO5: Awareness of computational issues, resources in distributed environment.

Suggestive list of experiments:-

1) Comparative Study of different operating systems
2) Demonstration of multitasking concept.
3) Implementing various process creation algorithms(FCFS,SJF and Round-Robin Scheduling)
4) Implementation of memory allocation policies.
5) Implementing Page replacement algorithms(FIFO,LIFO)
6) Implementing segmentation algorithms
7) Implementing file-handling algorithms
8) Implementing file-handling algorithms
9) Implementing file-handling algorithms
10) Demonstration of working of distributed OS environment.

Special Instructions if any: Nil

Designed by

- Mr. Vivek Kshirsagar
- Mrs. Meghana B. Nagori
- Mrs. Vijayshri A. Injamuri

CS3008: Lab Data Mining & Warehousing

Teaching Scheme		**Evaluation Scheme**	
Practical	2 Hrs/Week	Team Work	25 Marks
Credits	1	Practical /Viva-voce	25 Marks

Total Hours required for this practical course: 30 Hours.

Prerequisites:

Course Outcome:

After completion of this course student will be able to

CO1: Demonstrate the use of weka tool

CO2: Identify big data analytics domains and collect relevant data for analysis.

CO3:Awareness of various performance metrics for evaluation of data mining techniques

CO4: Effective Presentation of solutions to problems by choosing appropriate visualization tools.

CO5:Design a application from any suitable domain by incorporating all the core concepts

Suggestive list of programs.

1 .Installation and Comparative analysis of various mining tools.

2.Implement OLAP operations and schema using MS-Access and MS-Excel

3. Construct .arff and .csv file and import them in Weka.

4. Explore different visualization tools.

5. Explore different pre-processing options of WEKA tool.

6.Demonstrate the concept of association rule mining with Apriori algorithm

7.Implement the following clustering techniques:-

1.Hierarchical Clustering

2.K-means Clustering

8.Implement the following classification algorithms:-

1.J48

2.CART

3.Random tree

9.Demonstrate statistical analysis with Naïve-Bayes Classifier

10.Evaluation of data-mining techniques with different performance metrics

11. Demonstrating Text mining by selecting appropriate data mining technique.

12.Mini Project by performing the following on any domain of the students choice:-

Schema design

OLAP operations

Preprocessing of data

Application of suitable data mining technique

Evaluation by various performance metrics

Presentation of results by using visualization technique.

Special Instructions if any: Nil

Designed by

- Mrs. Meghana B. Nagori
- Mrs. Pallavi V. Kulkarni
- Mrs.Vrushali Chakkarwar

CS3009: Mini Project

Teaching Scheme			Evaluation Scheme	
Practical	4 Hrs/Week		Term Work	25 Marks
Total Credits	2		Practical/Viva-voce	25 Marks

Total Hours required for this practical course: 30 Hours.

Course Outcomes:
After completing this course, students will able to:

CO1: Identify area of interest / problem domain.

CO2: Collect related latest standard research papers and analyze them.

CO2: Use required Toolbox.

CO3: Apply the functions to solve problem / implement algorithm.

CO5: Develop solution and test it

Suggestive List in SDLC:

1. **Requirement Gathering and Planning phase for project.**
2. **Analyzing the requirement and resources of project.**
3. **Designing the architectural model for project.**
4. **Actual development of project.**
5. **Testing of project.**

The Software Development Lab consist of various stages for developing the software project, similarly SDL lab will cover all the process related activity of Software Engineering, Under this lab project should develop on one the following technology.

I. Cloud Computing
II. Android Based Application
III. Computer Vision System Toolbox
IV. Image Processing Toolbox
V. DSP System Toolbox
VI. Parallel Computing Toolbox
VII. Neural Network Toolbox
VIII. Bioinformatics Toolbox

Study of various MatLab/SciLab Tool Boxes related to Computer Science & Engineering Suggestive List is

1. **Computer Vision System Toolbox**
2. **Image Processing Toolbox**
3. **DSP System Toolbox**
4. **Parallel Computing Toolbox**
5. **Neural Network Toolbox**
6. **Bioinformatics Toolbox**

A mini project in a group of 2 to 3 students based on required tool boxes should be completed and a record for the same shall be submitted.

Internal Practical examination will consist of an oral examination.

1. The assessment in the oral examination
2. Record of programs/ projects submitted by the candidate.

Special Instructions if any: Nil

Designed by
All Faculty

Semester II

CS 3010 : Computer Networks

Teaching Scheme		**Evaluation Scheme**	
Lectures	3Hrs/Week	Test 1	15 Marks
		Test 2	15 Marks
		Teacher Assessment	10 Marks
Total Credits	3	End-Semester Examination	60 Marks

Total Hours required for this practical course: 30 Hours.

Prerequisite: CS2008: Data Structures

Course Description:

This course introduces the architecture, functions, components, and models of the computer networks and Internet. The principles of IP addressing and fundamentals of Ethernet concepts, media, and operations are introduced. Students will be able to build simple LANs, perform basic configurations for routers and switches, and implement IP addressing schemes.

Course Educational Objectives:

- Demonstrate knowledge of principles of computer networking
- Understand details and functionality of layered network architecture
- Know Internet applications and their protocols
- Understanding applications (e.g. Client Server applications, Web Services)
- Describe and use of Multimedia Information

Course Outcomes Expected:

After completion of this course student will be able to

CO1: Describe various protocols, models in Networks
CO2: Comprehend Network hardware, Media Types (cables, Wireless),
CO3: Compare UTP, Connectors, and Network interface Card
CO4: Design, implement and analyze simple computer networks.
CO5: Apply the different strategies Operations of TCP/UDP, FTP, HTTP, SMTP, SNMP

Detailed Syllabus:

UNIT-1 Introduction

Overview of computer network, Network hardware and software, Reference model- OSI and TCP/IP and their comparison Network layer- Network layer design issues, various routing Algorithms and congestion control algorithms

UNIT-2 TCP/IP

TCP/IP architecture, the internet protocols, IPv6, DHCP and Mobile IP, IP addressing, OSPF and BGP, multicast routing, the network layer in ATM networks

UNIT-3 Transport layer

The transport services, elements of transport protocols, internet, Transport protocols, ATM –AAL layer protocols, Performance issues

UNIT-4 The Application layer

Network security – principle of cryptography, secret key and public key algorithm, digital signature, Domain name system-The DNS name space, resource records, name server, simple network management Protocol –SNMP model, Electronic, mail- architecture and services, Message formats and message transfer, email privacy

UNIT-5 Multimedia Information and Networking

Lossless data compression, Video on Demand, Transmission in ATM network, Communication satellites.

TEXT AND REFERENCE BOOKS

1. Andrew .S. Tenenbaum, Computer networks, PHI
2. Alberto,Leon Garcia and Indrawidjaja, Communication networks- Fundamental concepts and key architectures, Tata mc-graw hill

Special Instructions if any: Nil

Designed by

- Mr. Sudhir G. Shikalpure
- Mr. Vivek P. Kshirsagar
- Mr. Prashant pathak

CS3011: System Programming and Compiler Construction

Teaching Scheme		**Evaluation Scheme**	
Lectures	3 Hrs/Week	Test 1	15 Marks
		Test 2	15 Marks
Tutorials	1Hrs/Week	Teacher Assessment	10 Marks
Total Credits	4	End-Semester Examination	60 Marks

Total Hours required for this course: 60 Hours.

Prerequisites: CS2003: Discrete Mathematical Structure
CS3003: Theory of Computation

Course Description: This course gives the introduction to system programming and compiler construction. It also gives the knowledge role of a lexical analyzer, specification of tokens, recognition of tokens, Lexical analyzer generator LEX, role of parser, context free grammars, eliminating ambiguity, eliminating left recursion, Top-Down parser. This course also gives the idea about Syntax Directed Translation and Intermediate Code Generation using different technique such as DAG, Three address code, etc. At the end this course gives the information runtime environment and issues in code generation.

Course Educational Objectives:

- Describe the utility of different system programs & system tools.
- Familiarize with the tradeoffs between run-time and compile-time processing (Linking & Loading techniques).
- Explore the use of compiler with its phases.
- Use of Syntax directed scheme for intermediate code generation.
- Construct & use of different compiler tools as LeX, Yacc for code generation & optimization.

Course Outcomes Expected:

After completion of this course student will be able to

CO1: Organize the functionalities & components of system software & tools into different layers for efficient code generation.

CO2: Apply the knowledge & technique to develop solutions to real world problems by compiling application programs.

CO3: ability to identify, formulate, and solve computer engineering problems with proper systematic & semantic approach.

CO4: Develop possible program constructs for further code generation with Type checking & memory management strategy

CO5: Design a simple compiler with tools & different with optimized techniques

UNIT-1 Introduction to System programming.

Components of System s/w , Language processors, Assemblers, Macro processor , Compilers, Interpreters, Device drivers.

UNIT-2 Loaders and Linkers.

Loaders:- Basic loader functions ,Compiler & go loader, design of absolute loader , design of direct linking loader.

Linker:- Linking & relocation concept, design of a linker ,static & dynamic linking loader, self relocating programs, editors.

UNIT-3 Lexical Analyzer and Syntax Analyzer.

Lexical Analyzer:- Role of a lexical analyzer, specification of tokens, recognition of

tokens, Lexical analyzer generator LEX.
Syntax Analyzer:- Role of Parser, Context free grammars, eliminating ambiguity, eliminating left recursion, Top Down parser(recursive descent parsing & no recursive descent parsing, Bottom Up parser, handle pruning, Shift reduce parsing, LR parser, LR parsing algorithm, SLR,LALR parser, Parser generators (YACC & BISON)

UNIT-4 Syntax Directed Translation and Intermediate Code Generation.

Syntax Directed definition, construction of syntax trees, dependency graph S-attributes & L-attributes evaluation, Symbol table structure, attributes & management.
Intermediate language structure, DAG's Three address code, intermediate code for Boolean expression, procedure calls ,control stmt's', Switch stmt's, Type checking

UNIT-5 Runtime Environment and Code Generation.

Static & Dynamic storage allocation, Introduction to garbage collection, parameter passing, returning values.
Code generation:- Issues in code generation, target language, Basic blocks, Flow graphs, Simple code generator, optimization of basic blocks, peephole optimization, machine dependent & Independent optimization.

TEXT BOOKS

1. A.V. Aho,Ullman&Sethi "Compilers –Principlers, tools & technique" Dragon book. Pearson.
2. D.M.Dhamdere " System programming & Operating systems"TMH

REFERENCE BOOKS

3. John.R Levine " Linkers &Loaders"Morgan Kaufman
4. John.R Levine ,T.mason, D brown " Lex &Yacc ", O Reilly,2000
5. S.Chattopadhya, " Comiler design" PHI
6. Andrew w.Appel, "Modern Compiler Implementation in C/Java",Cambridge university Press.

Special Instructions if any: Nil

Designed by

- Mrs. Arjumand M. Khan
- Mr. Vikul J. Pawar
- Mrs. Pallavi Kulkarni

CS3012: Advanced Algorithms

Teaching Scheme		Evaluation Scheme	
Lectures	3 Hrs/Week	Test 1	15 Marks
		Test 2	15 Marks
Tutorial	1	Teacher Assessment	10 Marks
Total Credits	3	End-Semester Examination	60 Marks

Total Hours required for this course: 60 Hours.

Prerequisites: CS2003: Discrete Mathematical Structures ,CS2008:Data Structures, CS1001:Basics of Computer & Information Technology, CS3001:Design and Analysis of Algorithm, CS3003:Theory of Computation

Course Description: This course introduces different design paradigms for designing efficient and optimal solutions to real world problems. The course will identify the computational issues in choosing an suitable algorithm for applying to solve complex problems. In addition to introducing the concept of NP Hard and NP Complete problems the course also exposes to geometrical algorithms and algorithms for big data sets.

Course Objectives:

- To identify issues related to the definition, creation and usage of classes, objects and methods.
- To discuss the principles of inheritance and polymorphism and demonstrate through problem analysis assignments how they relate to the design of methods, abstract classes and interfaces.
- To provide the foundation of good programming skills by discussing key issues to the design of object-oriented software, including programming design patterns.

Course Outcomes :

After completion of this course students will be able to:

CO1:Analyze the pros and cons of applying the different design paradigms in different Contexts.

CO2:Exposure to randomization as a tool for developing algorithms.

CO3:Relevance of analysis to the design of efficient computer algorithms.

CO4 :Identify the computational issues and apply suitable algorithms to solve it effectively

CO5:Conceptualize and design efficient and effective algorithmic solutions for different real-world problems.

Detailed Syllabus:

UNIT 1 **Theory of NP- Hard and NP-Complete Problems**

P, NP and NP-Complete complexity classes; Introduction to a few NP-Completeness proofs; Other complexity classes

Unit 2 **Approximation Algorithms**

Introduction, Combinatorial Optimization, approximation factor, PTAS, FPTAS, A few examples of approximation algorithms, Analysis of the expected time complexity of the algorithms.

Unit 3 **Probabilistic Algorithms & Randomized Algorithms**

Numerical probabilistic algorithms, Las Vegas and Monte Carlo algorithms, min-cut, markov and hidden markov models

Unit 4 **Computational Geometry**

Range Trees, Prority Search Trees, Quadtrees and k-d Trees, Convex Hulls

Unit 5 **Algorithms for large data sets**

Streaming algorithms,nearestneighbour search algorithms

TEXT BOOKS:

1. Introduction to Algorithms : T.H. Cormen, C.E.Leiserson and R.L. Rivest

2. Fundamentals of Algorithmics : G.Brassard and P.Bratley

REFERENCE BOOK:

1. Approximation Algorithms: Vijay V.Vazirani
2. Randomized Algorithms: R. Motwani and P.Raghavan

1) Special Instructions if any: Nil

Designed by

- Mrs. Meghana B. Nagori
- Mr. Sudhir G. Shikalpure
- Mrs. Pallavi V. Kulkarni

CS 3013: Data Science

Teaching Scheme		Evaluation Scheme	
Lectures	3 Hrs/Week	Test 1	15 Marks
		Test 2	15 Marks
Tutorial	1	Teacher Assessment	10 Marks
Total Credits	4	End-Semester Examination	60 Marks

Prerequisite: CS2002:Database Management Systems, CS3005: Data Mining & Warehousing

Course Objectives:

1. Identify and explain fundamental ingredients that constitute a recommendation engine.

2. Describe what data science is and the skill sets needed to be a data scientist.

Course Outcomes:

1. Describe the data science process and how its components interact.

2. Use API's And other tools to scrap the web and Collect data.

3. Build Your Own Recommendation system using existing Components

4. Work effectively in teams of data science projects.

5. Reason around ethical and privacy issues in data science conduct and apply ethical practices

Detailed Syllabus:

Unit 1	**Introduction to data science** Big Data and Data science Hype, Datafication, The emergence of data lakes, Introduce the federation business data lakes, data lake ingests ,stores and processes data
Unit 2	**Recommendation systems** Algorithmic ingredients of a recommendation Engine, Dimensionality Reduction, Singular Value Decomposition, Principal Component Analysis, Building your own recommendation systems.
Unit 3	**Mining Social Network Graphs:** Social Networks as graphs, clustering of graphs, direct discovery of communities in graphs, partitioning of graphs, neighborhood properties in graphs.
Unit 4	**Data Science and Ethical Issues** Discussions on privacy, security and ethics
Unit 5	**Data Wrangling and Forecasting** Web Scraping and API's , Compare various types of time-series components, Discuss Time

	Series, Forecasting methods(ARIMA model),Evaluation Criteria of Forecasting Models.

CS 3014: LAB: Application Development Tools

Teaching Scheme	**Examination Scheme**
Practical:2 Hrs/Week	**Term Work : 25 Marks**
Credits : 1	**Practical /Viva : 25 Marks**

Prerequisites: Lab Web Technology

Course description: This course aims at developing skill amongst students in area of Web Based application development using ASP.NET, C#, Ajax etc.

Course Objective

1. Describe .NET Framework and introduce its IDE to create Presentation Layer
2. Describe basic concepts in C# and introduce important file and folders in C#
3. Enable Students on how to use Standard and Validation Controls
4. Demonstrate on various ways of interacting with Database
5. Demonstrate use of AJAX and Web Services

Course Outcome

After completion of course students will be able to

1. Develop presentation layer for their web application
2. Differentiate and Use proper files and folders to address situation
3. Implement the Standard and Validation Controls
4. Connect with Database and Manage it
5. Implement AJAX and Web Services

Unit1

Getting started with .NET: Introduction to .NET Framework and Visual Studio.NET,Kind of Applications that can be developed using Visual Studio.NET, Website v/s Web Application, Creating a new sample Web Project with Visual Studio

Exploring Visual Studio IDE: Solution Explorer, Properties Window, Tool Box, Server Explorer etc.

Unit 2

Introduction to C#: Working with Variables, Data Types, Data Type Conversion, Operators and Expressions, Creating Classes and Objects in C#, Using Namespaces, Arrays, Exception Handling in C#, Navigating amongst Web Pages, Event Handling

Important Files and Folders in Web Application: All System Folders, Web.Config, Global.asax, Building sites with Master Pages, Using User Controls

Unit 3

Building ASP.NET Pages: Standard Controls, Validation Controls,

State Management: ASP.NETPageLife Cycle, Session Management, Managing Query String, View State in C#

Unit 4

Working with Data: ADO.NET Architecture, Connected & Disconnected Architectures, SQL Connection, SQL Command & important Classes for operating database related operations (CRUD), Using Datasets& Data Adapters

Working with Data Controls: Grid View, Repeater Control
Unit 5
AJAX: Ajax Architecture, Script Manager, Update Panel, Ajax Control Toolkit
Web Services: Creating and Consuming Web Services
Deploying ASP.NET Websites: Installing and configuring website using IIS
Introduction to MVC Architecture

Text/Reference Books

1. The Complete Reference ASP.NET
2. The Complete Reference C#

Suggestive List of Programs

1. To Study the ASP.Net Framework
2. To Study &Create Presentation Layer using HTML, CSS& ASP.NET Controls
3. To Study &Create Master Page, User Control etc
4. To implement OOP concepts in C#
5. To Study& Use Validation Controls in ASP.NET
6. To Study, Create and Connect with Database using ASP.NET & SQL Server
7. To Study& ImplementWeb Services
8. To Study &Implement AJAX in ASP.NET
9. To Study & DeployProject on IIS
10. Mini Project

List of Experiments

Sr. No.	Details
1	To Study the ASP.Net Framework
2	To Study & Create Presentation Layer using HTML & CSS
3	To Study & Create Master Page, User Control etc
4	To Study & Use Standard Controls in ASP.NET
5	To Study & Use Validation Controls in ASP.NET
6	To Study, Create and Connect with Database using ASP.NET & SQL Server
7	To Study & Implement Web Services
8	To Study & Implement AJAX in ASP.NET
9	To Study & Deploy Project on IIS
10	Mini Project

Special Instructions if any: Nil

Designed by

- Mr. Prashant D. Pathak
- Mr. Vikul J. Pawar

CS 3015: Lab Computer Network

Teaching Scheme		**Evaluation Scheme**	
Practical	2Hrs/Week	Term Work	25 Marks
Credit	1	Practical/Viva-voce	25 Marks

Prerequisites: CS2013: Lab Data structures

Course Outcomes:

After completion of this course student will be able to

CO1	Recognize the different internetworking devices and their functions.
CO2	Role of protocols in networking.
CO3	Design and apply subnet masks and addresses to fulfill networking requirements.
CO4	Features of TCP/IP Protocol
CO5	Analyze the featuresand operations of various application layer protocols such as Http, DNS

The term work shall consist of following practical

1	Introduction to Networking Devices
2	Understanding / Illustrate the network features of peer to peer network .
3	Understanding / Illustrate the network features of Client Server network .
4	Build a Category 5 or Category 6 Unshielded Twisted Pair (UTP) Ethernet crossover cable
5	Connecting 2 Computers together using a Crossover cable
6	Configure TCP/IP in LAN
7	File Transfer / Sharing/ Virtual Desktop Access
8	Study of basic network command and Network configuration commands.
9	Program for simple RSA algorithm to encrypt and decrypt the data.
10	Client/Server chat application

Special Instructions if any: Nil

Designed by

- Mr. Sudhir G. Shikalpure
- Mr. Vivek P. Kshirsagar
- Mr. Prashant Pathak

CS3016: Lab Advanced Algorithms

Teaching Scheme		**Evaluation Scheme**	
Practical	2 Hrs/Week	Team Work	25 Marks
Credits	1	Practical /Viva-voce	25 Marks

Total Hours required for this practical course: 30 Hours.

Prerequisites: Lab Programming Language, Lab Data Structure, Lab Object Oriented Programming, Lab Design and analysis of Algorithm , Lab Programming in Java,

Course Outcome:

After completion of this course student will be able to:-

CO1: Identify Data Structures, Design paradigms and Computational complexity in the design of simple tools
CO2: Demonstrate relationships among NP-Complete Problems
CO3: Implement the approximate algorithms approach to solve some NP-Complete Problems.
CO4: Demonstrate randomness by solving some examples
CO5: Implement algorithms for geometry and large data-sets.

Suggestive list of experiments:-

1) Design of simple tool of choice for revising of basic concepts.
2) Implement program to show relationship between clique,vertex-cover and independent set.
3) Implement Approximate algorithm technique for vertex cover/steiner tree.
4) Implement approximate algorithms on various variants of Travelling Salesman problem.
5) Demonstrate randomness by implementing Quicksort algorithm.
6) Demonstrate randomness by implementing min-cut algorithm.
7) Demonstrate with a Program the Markov property and stationery markov chain
8) Implementing the Viterbi algorithm
9) Implementing the forward algorithm
10) Implementing algorithms from geometry problems and large data sets.

Special Instructions if any: Nil

Designed by

- Mrs. Meghana B. Nagori
- Mr. Sudhir G. Shikalpure
- Mrs. Pallavi V. Kulkarni

CS3017: Lab: Data Science

Teaching Scheme:	Examination Scheme:
Practical: 2 Hrs/Week	Term Work : 25 Marks
Credit :1	

Course Outcome:

CO1: Use Python and other tools to scrape, clean, and process data
CO2: Use data management techniques to store data locally and in cloud infrastructures
CO3: Use statistical methods and visualization to quickly explore data

Minimum of 8 Programs should be completed which will be based on the subject and record for the same shall be submitted

Case Study of real time problems such as
From any of the following Domains
Health Care, Finance, It for Analytics, Marketing Analytics & Decision & Operations Analytics.

CS3018: Seminar

Teaching Scheme			Evaluation Scheme	
Practical	2 Hrs/Week			
			Term Work	25 Marks
Total Credits	1			

Total Hours required for this practical course: 30 Hours.

Lab Outcomes:

After completing this course, students will able to:
CO1**:** Collect Organize & Analyze information about emerging technologies /market demands/current trends.
CO2: Exhibit effective communication skills, stage courage, and confidence.
CO3: Demonstrate intrapersonal skills,
CO4: Awareness in keeping with new innovations and inventions.
Guidelines for presenting a seminar:

- The seminar will consist of a typewritten report covering the topic related to his area of final year project.
- If more number of students are working on same project then they should separate the seminar topics from the project area which is relevant and which will contribute for completion of project.

- Weekly report of students work for finalization of his area of work and topic of seminar should be submitted to the faculty during designated hours meant for seminar
- Format of weekly report should be finalized by the department with sufficient inputs received from the students. It should have following stage wise reports:

  Project Area and Project Groups by 3rd week

  Tentative seminar topics by 4th week

  Literature/Field Study Mechanism identified sources and strategy by 5th week

  Weekly report on Literature/Field Study 6th, 7th& 8th week

  Presentation Format contents & Trial Presentations to student groups 9th, 10th week

  Journal on above stages and Final Presentation Report 11th week

- It is expected that the candidate prepares a report based on outcomes of literature studies, field visits, observation schedules, focus group meetings etc related to a problem in relevant technology area.
- The report shall be tested for any plagiarism out of books, journals and internet based articles and reports by appropriate web based tool.
- The candidate shall deliver seminar on the topic on first two occasions to students of his class for peer assessment.
- Format for peer group assessment should be designed by the faculty with approval of department. Peer assessment should not be given more than 15% weightage
- Final presentation for term work should be attended by minimum TWO faculty members. Each candidate may be given time minimum of 8 to 10 minutes.
- Assessment criteria for seminar delivery for term work should be designed by the faculty with inputs received from students of the class. It should include provision for peer group assessment as per the norm stated above.
- Assessment Criteria so designed will be displayed on the department notice board with the approval from department along with these guidelines.

Special Instructions if any: Nil

Designed by
All Faculty