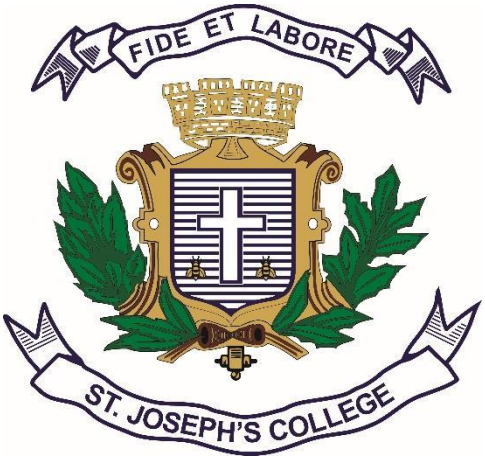
ST. JOSEPH'S COLLEGE (AUTONOMOUS)

BENGALURU-27



Re-accredited with **'A++' GRADE with 3.79/4 CGPA** by NAAC Recognized by UGC as College of Excellence

DEPARTMENT OF COMPUTER SCIENCE

SYLLABUS FOR POSTGADUATE PROGRAMME

For Batch 2021-2024

SUMMARY OF CREDITS

FIRST SEMESTER

THEORY						
Code	**Title**	**Hours Per Week**	**Credits**	**IA**	**SE**	**Total**
CS7121	Object Oriented Programming using JAVA	4	4	30	70	100
CS7221	Theory of Computation	4	4	30	70	100
CS7321	Design and Analysis of Algorithm	4	4	30	70	100
CS7421	Cyber Security	4	4	30	70	100
PRACTICAL						
MCS1P1	Object Oriented Programming JAVA Lab	6	3	30	70	100
MCS1P2	Design and Analysis of Algorithm Lab	6	3	30	70	100

Total Number of Credits: 22

SECOND SEMESTER

THEORY						
Code	**Title**	**Hours Per Week**	**Credits**	**IA**	**SE**	**Total**
CS8121	Advanced Database Management System	4	4	30	70	100
CS8221	Machine Learning with Python	4	4	30	70	100
CS8321	Advanced Web Technologies	4	4	30	70	100
CS8421	Principles of Compiler Design	4	4	30	70	100
CS8521	Software Project Management	4	4	30	70	100
PRACTICAL						
MCS2P1	ML with Python lab	6	3	30	70	100
MCS2P2	Advanced Web Technologies &ADBMS Lab	6	3	30	70	100

Total Number of Credits: 26

THIRD SEMESTER

THEORY						
Code	**Title**	**Hours Per Week**	**Credits**	**IA**	**SE**	**Total**

CS9121	**Data Analytics with HADOOP**	**4**	**4**	**30**	**70**	**100**
CS9221	**Mobile Communication and Applications**	**4**	**4**	**30**	**70**	**100**
Department Elective (One of the two)						
CSDE9321	**1) Internet of Things**	**4**	**4**	**30**	**70**	**100**
CSDE9421	**2)Cloud Computing and Information Storage Management**	**4**	**4**	**30**	**70**	**100**
Open Elective (For other students)						
CSOE9121	**Web Technologies**	**2**	**2**	**15**	**35**	**50**
PRACTICAL						
MCS3P1	**Mobile Applications Lab**	**6**	**3**	**30**	**70**	**100**
MCS3P2	**Database Applications Development Lab**	**6**	**3**	**30**	**70**	**100**

Total Number of Credits: 20

FOURTH SEMESTER

THEORY						
Code	**Title**	**Hours Per Week**	**Credits**	**IA**	**SE**	**Total**
CS0121	**Image Processing**	**4**	**4**	**30**	**70**	**100**
CS0221	**Advanced Operating System**	**4**	**4**	**30**	**70**	**100**
PRACTICAL						
MCS4P1	**Major Project /Internship**	**24**	**12**	**90**	**210**	**300**

Total Number of Credits: 20

KEY WORDS: DE – Departmental Elective and OE – Open Elective

CORE COURSES (CC)	
Course Title	**Code Number**
Object Oriented Programming using JAVA	**CS7121**
Theory of Computation	**CS7221**
Design and Analysis of Algorithm	**CS7321**

Cyber Security	**CS7421**
Advanced Database Management System	**CS8121**
Machine Learning with Python	**CS8221**
Advanced Web Technologies	**CS8321**
Principles of Compiler Design	**CS8421**
Software Project Management	**CS8521**
Data Analytics with HADOOP	**CS9121**
Mobile Communication and Applications	**CS9221**
Image Processing	**CS0121**
Advanced Operating System	**CS0221**

DISCIPLINE SPECIFIC ELECTIVE COURSES (DSE)	
Course Title	**Code Number**
1) Internet of Things	**CSDE9321**
2)Cloud Computing and Information Storage Management	**CSDE9421**

GENERIC ELECTIVE COURSES (GSE)/ Can include open electives offered	
Course Title	**Code Number**
Web Technologies	**CSOE9121**

SKILL ENHANCEMENT COURSE (SEC) – **Any practical oriented and software based courses offered by departments to be listed below**	
Course Title	**Code Number**
Object Oriented Programming JAVA Lab	**MCS1P1**

Design and Analysis of Algorithm Lab	**MCS1P2**
ML with Python lab	**MCS2P1**
Advanced Web Technologies &ADBMS Lab	**MCS2P2**
Mobile Applications Lab	**MCS3P1**
Database Applications Development Lab	**MCS3P2**
Major Project /Internship	**MCS4P1**

VALUE ADDED COURSES (VAC) **Certificate courses that add value to the core papers can be listed**	
Course Title	**Code Number**
Cyber security and Ethical Hacking	
Web Design	

Online courses offered or recommended by the department to be listed	
Course Title	**Code Number**
Python for Data Science (NPTEL online courses)	
Block chain Architecture Design and use cases (NPTEL online courses)	
Google Cloud Computing Foundation Course(NPTEL online courses)	
Data Analytics with Python(NPTEL online courses)	

Question Paper pattern:

Part A

Multiple choice Questions 15 questions each carries one mark (15 * 1=15 marks)

Part B

Internal Choice Questions from all the 5 units (5 * 5 =25 marks)

Part C

Three questions out of four each carries 10nmarks (3 * 10=30marks)

Course Outcomes and Course Content

Semester	I
Paper Code	**CS7121**
Paper Title	**Object Oriented Programming Using Java**
Number of Teaching hours per week	04
Total number of teaching hours per semester	60
Number of credits	04

Objective of the Paper:

To introduce the concepts and principles of Java Programming language and to design and implement object oriented solutions to the simple and complex problems. Also to give students experience in Java Programming and GUI application design with data base.

Unit 1 **12Hrs**

Introduction to Object Oriented Programming and Classes:

Introduction to Object Oriented Programming- Object-Oriented Programming(OOP) Principles- The Evolution of Java- Declaring Objects - Introducing Methods - Overloading methods – Constructors - Parameterized Constructors - this Keyword. Garbage Collection - the finalize () Method - Introducing Access Control - Understanding static - Introducing nested and inner classes - String class - String Buffer Class - Command Line Arguments

Unit 2 **12Hrs**

Inheritance in Java

Inheritance Basics - Multilevel Hierarchy- Using super - Method overriding - Abstract keyword- Using final with inheritance.

Interfaces and Packages

Defining Interfaces - Implementing Interfaces - Extending Interfaces, Creating Packages - CLASSPATH variable - Access protection - Importing Packages.

Exception Handling in Java

Try-catch-finally mechanism - throw statement - throws statement - Built-in-Exceptions – Custom Exceptions.

Unit 3 **12Hrs**

Multithreading, Generics and The Collections Framework

Java Thread Model - Life cycle of a Thread - Java Thread Priorities - Runnable interface and Thread Class- Thread Synchronization – Inter Thread Communication.

The Collections Framework

The Collections Overview – Collection Interface – List Interface – Set Interface – SortedSet Interface – Queue Interface - ArrayList Class – LinkedList Class – HashSet Class – Using an Iterator – The For Each Statement

Unit 4 **12Hrs**

Introducing GUI Programming with Swing

Swing Basics – Components and Containers – JLabel and ImageIcons- JTextField – Swing Buttons – JTabbedPane – JScrollPane – JList – JComboBox – JTable – Swing Menus

Event Handling

Delegation EventModel- Event Classes – Key Event Class – Event Listener Interface –AdapterClasses

Unit 5 **12Hrs**

DATABASE PROGRAMMING USING JDBC

Introduction to JDBC, JDBC Drivers and Architecture, Connecting to and querying a database–Automatic driver recovery- -Creating a Statement for executing query-Executing a query-Processing a Query's Result Set

NOTE: 6 hours of self-study will be assigned from the above units.

REFERENCES

- Herbert Schildt, "Java the Complete Reference", 11thEdition, McGraw-Hill Osborne Media.
- Cay S. Horseman and Gary Cornell, "Core Java, Vol.2: Advanced Features", 8th Edition, Prentice Hall.
- Computer Bible Games with Java 11th Edition: A Java JFC Swing GUI Game Programming
- Beginning Java 8 APIs, Extensions and Libraries: Swing, JavaFX, JavaScript, JDBC and Network Programming APIs (Expert's Voice in Java) 1st Edition

BLUEPRINT

Code number: **CS7121**

Title of the paper: **Object Oriented Programming Using Java**

Chapter	Number of Hours	Total marks for which the questions are to be asked (including bonus questions)
Unit I	12	14
Unit II	12	20
Unit III	12	20
Unit IV	12	16
Unit V	12	10
TOTAL	**60**	**80**
Maximum marks for the paper (Excluding bonus question)= 70		

Practical I

MCS1P1-OBJECT ORIENTED PROGRAMMING USING JAVA LAB
(11 sessions 6hrs/week)

Program List

1. Implement the concept of Overloading methods
2. Implement the concept of control statements and Arrays in the class
3. Implement the concept of class, data members, member functions and access specifiers.
4. Implement the concept of function overloading & Constructor overloading.
5. Implement the static keyword – static variable, static block, static function and static class
6. Implement String and String Buffer classes.
7. Implement this keyword and command line arguments.
8. Implement the concept of inheritance, super, abstract and final keywords in java.
9. Implement package and interface keywords in java
10. Implement Exception Handing in java
11. Implement multithreading – Thread class, Runnable interface, thread synchronization and thread communication.
12. Implement generic concept – generic class and generic interface
13. Implement collections – collection Interfaces and collection classes
14. Implement Swing components and containers
15. Implement Event Handling
16. Implement the operations in JDBC

Course Outcomes: At the end of the course, the student should

CO1	Knowledge	Have developed a good knowledge about Object Oriented Programming.
CO2	Understand	Have developed a very good understanding on Advanced concepts of Java Programming, syntax and programming conventions.
CO2	Apply	Be able to program and associate with various logics.
CO3	Analyze	Be able to compare and relate the complexity of coding with the knowledge of testing and debugging.
CO4	Evaluate	Be able to choose appropriate approach to solve the various problems.
CO5	Create	Be able to develop complete real world problems with appropriate Java coding knowledge.

Course Outcomes and Course Content

Semester	I
Paper Code	**CS7218**
Paper Title	**THEORY OF COMPUTATION**
Number of teaching hours per week	04
Total number of teaching hours per semester	60
Number of credits	04

Course Objectives

The Course is designed to give theoretical understanding of the subject from the perspective of formal languages and to lay foundations for Compiler Design and Concurrent design. To develop Problem Solving Ability. Through the contents of the course the students will be able to understand Automata Theory, Formal Languages, Computability Theory, Context Free Grammars and Various Machines.

UNIT-I

Introduction to set theory concepts and Automata Theory

a. Set Theory **2Hrs**

Introductory concepts of Set Theory

b. Automata Theory **6Hrs**

Introduction to Finite Automata, Definition of Alphabets, Strings, Languages, Sentences , Transition Functions and Extended Transition Functions, Notations to represent Automata

c. Introduction to Central Concept of Automata Theory **6Hrs**

Deterministic Automata, Non Deterministic Automata, Applications of Finite Automata, Understanding Mealy and Moore Machines

UNIT-II

Finite Automata Problem Solving and Regular Expressions

a. Problem Solving **8Hrs**

Problem solving in DFA and NFA, Conversions from NFA to DFA, Finite automata with Epsilon-transitions,

Conversions from Epsilon-NFA to DFA , Equivalence and Minimization of DFA

b. Regular Expressions **2Hrs**

Introduction to Regular Expressions and Regular Expressions, Applications of Regular Expressions

c. Regular Languages **2Hrs**

Proving languages not to be regular languages, Closure and Decisive properties of regular languages

UNIT-III

Context Free Grammar

a. Introduction to Grammar **4Hrs**

Introduction to Context Free grammars and Chomsky's Grammar Hierarchy, Derivation and Derivation Tree and Types of Derivation, Applications of Grammar and Ambiguity in grammars and Languages.

b. Various Representation of Grammar **8Hrs**

Elimination of useless symbols, Epsilon productions and Unit productions, Representation of CNF and GNF. Conversion to CNF and GNF.

c. Context Free Languages **2Hrs**

Normal forms for CFGs; The pumping lemma for CFGs; Closure properties of CFLs, Pumping lemma for CFLs and its use.Closure properties of CFLs.

UNIT-IV

Push Down Automata and its applications **10Hrs**

a. Introduction to Pushdown Automata

Definition of the Pushdown automata and 7 of PDA, Deterministic, nondeterministic and Instantaneous descriptions of PDAs, Language acceptance by final states and by empty stack. PDA Problems.

UNIT-V

Introduction to Computability Theory

a. Turing Machines **6Hrs**

The turning machine; Programming techniques for Turning Machines, Extensions to the basic Turning Machines, Turing Machine and Computers

b. Decidability **2Hrs**

Decidability and Halting Problems

c. Reducibility **2Hrs**

Undecidable Problems in Language Theory, Simple Undecidable Problem and Mapping Reducibility.

References

- Introduction to AutomataTheory Languages, and Computation, by J.E.Hopcroft, R.Motwani & J.D.Ullman (3rd Edition) – Pearson Education
- Theory of Computer Science (Automata Language & Computations), by K.L.Mishra& N. Chandrashekhar, PHI
- Cohen, "Introduction to Computer Theory", John Wiley
- "Finite Automata and Formal Languages", A.M.Padma Reddy
- M. Sipser, Introduction to Theory of Computation, PWS Publishing Corporation, 1997.
- T.C. Martin, Theory of Computation, Tata McGraw-Hill
- H.R. Lewis, C.H. Papadimitrou, Elements of the Theory of Computation, PHI.

BLUEPRINT

Code Number: **CS7218**

Title: **Theory of Computation**

Chapter	Number of Hours	Total marks for which the questions are to be asked (including bonus questions)
Unit I	14	20
Unit II	12	16
Unit III	14	20
Unit IV	10	14
Unit V	10	10
TOTAL	**60**	80
Maximum marks for the paper (Excluding bonus question)= 70		

Course Outcome: At the end of the course the student should

CO1	Knowledge	Identify key notions of computation, such as algorithm, computability, decidability, reducibility, and complexity, through problem solving.
CO2	Understand	Understand the models of computation, including formal languages, grammars and automata, and their connections.

CO3	Apply	Apply Mathematical Foundations and algorithmic principles in problem solving for modeling and designing of computer and software systems.
CO4	Analyze	Analyze and design Finite automata, Pushdown automata, Turing machines, Formal languages, and Grammars.
CO5	Evaluate	Solve Computational problems and able toprove the basic result of Theory of Computation
CO6	Create	Construct algorithms for different problems and argue formally about correctness on different restricted machine models of computation.

Course Outcomes and Course Content

Semester	I
Paper Code	**CS7321**
Paper Title	**DESIGN AND ANALYSIS OF ALGORITHMS**
Number of teaching hours per week	04
Total number of teaching hours per semester	60
Number of credits	04

Objective of the Paper:

This paper introduces the paradigms and approaches used to analyse and design algorithms and to appreciate the impact of algorithm design in practice. It introduces the classic algorithms in the various domains and the different techniques for designing efficient algorithms.

UNIT 1 **12 HRS**

INTRODUCTION
The Role of Algorithms in Computing. Analyzing Algorithms, Growth of functions using Asymptotic Notation, Recurrence relations. Design Methods: General Consideration, Algorithm design paradigms and representative problems.

UNIT 2 **12 HRS**

DIVIDE AND CONQUER
Binary search, Merge Sort, Quick Sort, Arithmetic with Large integers.

UNIT 3 **12 HRS**

GREEDY METHOD

Minimal Spanning Tree, Shortest Paths, Knapsack Problem. Dynamic Programming, Chained Matrix Multiplication, Optimal Storage on Tapes, Shortest Paths (Dijkstra's and Floyd– Warshall algorithm), Optimal Search Trees.

UNIT 4 **12 HRS**

BACKTRACKING METHOD

8-queens problem, Graph Coloring, Hamiltonian Cycles, Branch and Bound -0/1 Knapsack problem, Travelling Salesman problem, Approximation Graph Coloring, Task Scheduling, Bin Packing.

UNIT 5 **12 HRS**

GRAPH ALGORITHMS

BFS, DFS and its applications. Polynomial Evaluation, Intractable Problems: Basic Concepts, Nondeterministic Algorithms, NP Completeness, Cook's Theorem, Examples of NP-Hard and NP-Complete problems.

NOTE: 6 hours of self-study will be assigned from the above units.

REFERENCES:

- E. Horowitz and S. Sahani, Fundamentals of Computer Algorithms, Galgotia, New Delhi.
- Aho, J. Hopcroft and J.Ullman, The Design and Analysis of Computer Algorithms, Addison Wesley.
- S.E.Goodman and S.T.Hedetniemi, Introduction to the Design and Analysis of Algorithms, McGraw Hill.
- G.Brassard, and P.Bratley, Algorithmics, PHI.
- S.K.Basu, Design Methods and Analysis of Algorithms, PHI.

BLUEPRINT

Code number: **CS7321**

Title of the paper: **DESIGN AND ANALYSIS OF ALGORITHMS**

Chapter	Number of Hours	Total marks for which the questions are to be asked (including bonus questions)
Unit I	12	14
Unit II	12	20
Unit III	12	20
Unit IV	12	16
Unit V	12	10
TOTAL	**60**	**80**
Maximum marks for the paper (Excluding bonus question)= 70		

Practical II

MCS1P2-DESIGN AND ANALYSIS OF ALGORITHMS LAB

(11 sessions 6hrs/week)

Program List

1. Implementing Divide and Conquer MinMax Algorithm find the greatest and the smallest numbers from a given set of numbers.

2. Sort a given set of elements using the Quicksort method and determine the time required to sort the elements.

3. Implement Merge Sort algorithm to sort a given set of elements and determine the time required to sort the elements

4. Obtain the Topological ordering of vertices in a given digraph.

5. Compute the transitive closure of a given directed graph using Warshall's algorithm.

6. Implement 0/1 Knapsack problem using Dynamic Programming.

7. From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.

8. Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm.

9. Print all the nodes reachable from a given starting node in a digraph using BFS method.

10. Check whether a given graph is connected or not using DFS method.

Course Outcomes: At the end of the course, the student should

CO1	Knowledge	Demonstrate a familiarity with major algorithms and data structures**.**
CO2	Understand	Ability to understand how the choice of data structures and the algorithm design methods impact the performance of programs
CO2	Apply	Ability to choose appropriate algorithm design techniques for solving problems
CO3	Analyze	Ability to analyze the performance of algorithms
CO4	Evaluate	Be able to evaluate the algorithms based on its time and space complexity.
CO5	Create	Developing efficient algorithms for simple computational tasks

Course Outcomes and Course Content

Semester	I
Paper Code	**CS7421**
Paper Title	**Cyber Security**
Number of teaching hours per week	04
Total number of teaching hours per semester	60
Number of credits	04

Objective of the Paper:

This course deals with how to protect information and information infrastructure in cyberspace, build capabilities to prevent and respond to cyber threats, reduce vulnerabilities and minimize damage from cyber incidents through a combination of institutional structures and cyber laws.

.

UNIT-I

Cryptography **12 Hours**

Introduction to Cryptography, Symmetric key Cryptography, Asymmetric key Cryptography ,Convention Encryption Model, Classical Encryption Techniques, Simplified DES, Block Cipher Principles, The Data Encryption Standard, The Strength Of DES, Differential And Linear Cryptanalysis Block Cipher, Design Principles, Block Cipher Modes Of Operations, Conventional Encryption Algorithms, Public Key Encryption.

UNIT-2

Network Security **14 Hours**

Introduction To The Concepts Of Security, Security Approaches, Principles Of Security, Types Of Attacks, Intruders, (IDS and IPS)Intrusion Detection And Prevention System, Message Authentication, Digital Signatures, Applications of Cryptography. Overview of Firewalls- Types of Firewalls, User Management, VPN Security Protocols: - security at the Application Layer- PGP and S/MIME, Security at Transport Layer- SSL and TLS, Security at Network Layer-IPSec.

UNIT- 3

Introduction to Cyber Security **12 Hours**

Overview of Cyber Security, Security Fundamentals-:Authentication Authorization Accountability, Social Media, Social Networking and Cyber Security. Cyber Attack and Cyber Services, Computer Virus – Computer Worms – Trojan horse. Vulnerabilities - Phishing - Online Attacks – Pharming - Phoarging – Cyber Attacks - Cyber Threats - Zombie- stuxnet - Denial of Service Vulnerabilities - Server Hardening-TCP/IP attack-SYN Flood.

UNIT-4

Cyberspace and the Law **10 Hours**

Introduction, Cyber Security Regulations, Roles of International Law, the state and Private Sector in Cyberspace, Cyber Security Standards. The INDIAN Cyberspace, National Cyber Security Policy 2013.

UNIT-5

Introduction to Cyber Forensics **12 Hours**

Handling Preliminary Investigations, Controlling an Investigation, Conducting disk-based analysis, Investigating Information-hiding, Scrutinizing E-mail, Validating E-mail header information, Tracing Internet access, Tracing memory in real-time.

REFERENCES:

- Thomas R, Justin Peltier, John, Information Security Fundamentals, Auerbach Publications.
- AtulKahate, Cryptography and Network Security 2nd Edition, Tata McGrawHill.
- Stallings, “Cryptography & Network Security - Principles & Practice”, Prentice Hall, 3rd Edition.
- Nina Godbole, SunitBelapure, Cyber Security, Wiley India 1st Edition.
- Dan Shoemaker and Wm. Arthur Conklin, Cyber security: The Essential Body Of Knowledge, Delmar Cengage Learning; 1 edition

BLUEPRINT

Code number: **CS7421**

Title of the paper: **Cyber Security**

Chapter	Number of Hours	Total marks for which the questions are to be asked (including bonus questions)
Unit I	12	15
Unit II	14	20
Unit III	12	20
Unit IV	10	10
Unit V	12	15
TOTAL	**60**	**80**
Maximum marks for the paper (Excluding bonus question)= 70		

Course Outcomes: At the end of the course, the student should

CO1	Knowledge	Have developed a good knowledge about securing both clean and corrupted systems, protect personal data, and secure computer networks.
CO2	Understand	Have developed a very good understanding on key terms and concepts in cyber law, intellectual property and cyber-crimes, trademarks and domain theft.
CO2	Apply	Be able to Interpret and forensically investigate security incidents
CO3	Analyse	Be able to analyse and resolve security issues in networks and computer systems to secure an IT infrastructure.
CO4	Evaluate	Be able to design, develop, test and evaluate secure software.
CO5	Create	Be able to develop policies and procedures to manage enterprise security risks

Course Outcomes and Course Content

Semester	II
Paper Code	**CS8121**
Paper Title	**Advanced Database Management Systems**
Number of Teaching hours per week	04
Total number of teaching hours per semester	60
Number of credits	04

Objectives

This paper covers database design and SQL; it also provides an understanding of relational Database, Parallel Database and Data Warehouse. It emphasis on a practical approach to design a Complete database for various entities.

UNIT I

Database Concepts:

a. Introduction to database **5 Hrs**

Characteristics of Database Approach, Functional Dependency, Normalization.

b. Query Processing and Optimization **5 Hrs**

Introduction, General Optimization, Strategies, Algebraic Manipulation-Algorithm for optimizing Relational Expressions.

c. Hands on practice **2 Hrs**

All SQL commands with certain entities. (Bank, Education Institution, Library Management System etc.)

UNIT II

Transaction Processing and Concurrency Control:

a. Introduction to transaction processing and concurrency control **6 Hrs**

Definition of Transaction and ACID properties, transaction Recovery, Concurrency Control Techniques: Lock based Concurrency control -Optimistic Concurrency ControlTimestamp based Concurrency Control, Deadlocks.

b. Database Security **2 Hrs**

Security Issues, Control Measures- Discretionary, mandatory and role based access control.

c. Database Recovery Techniques **2 Hrs**

Recovery Concepts- Deferred Update and Immediate Update techniques, Shadow Paging – ARIES.

d. Hands on practice **2 Hrs**

Implementing transaction recovery on a database.

UNIT III

PARALLEL DATABASES:

a. I/O Parallelism **5 Hrs**

Inter and Intra Query Parallelism – Inter and Intra operation Parallelism Design of Parallel Systems.

b. Distributed Database Concepts **5 Hrs**

Distributed Data Storage – Distributed Transactions – Commit Protocols – Concurrency Control – Distributed Query Processing.

c. Case Study **2 Hrs**

Comparative study between inter and intra operation parallelism (Case Study)

UNIT IV

DATA WAREHOUSE FUNDAMENTALS:

a. Introduction to Data Warehouse **6 Hrs**

OLTP Systems, Differences between OLTP Systems and Data Warehouse, Characteristics of Data Warehouse, Functionality of Data Warehouse, Advantages andApplications of Data Warehouse.

b. Methodology **6 Hrs**

Top- Down and Bottom-Up Development Methodology, Tools for Data warehouse development, Data Warehouse Types.

UNITV

Data Warehouse Architecture:

a. Introduction **4 Hrs**

Components of Data warehouse Architecture, Dimensional Modeling- Introduction, E-R Modeling, E-R Modeling VS Dimensional Modeling.

b. Data Warehouse Schemas **4 Hrs**

Star Schema, Inside Dimensional Table, Inside Fact Table, Fact Less Fact Table, Granularity, Star Schema Keys, Snowflake Schema, Fact Constellation Schema.

c. Data Warehouse & OLAP **4 Hrs**

Introduction to OLAP, Characteristics of OLAP, Steps in the OLAP Creation Process, Advantageous of OLAP, What is Multidimensional Data.

d. OLAP Architectures **4 Hrs**

MOLAP, ROLAP, HOLAP, Data Warehouse and OLAP-Hypercube & Multi cubes.

Self Study 6 hrs:

- Data Models, Schemas, Three Schema Architecture and Data Independence; Database Design: ER Modeling – ER diagrams.
- Dimensional Model with Examples.

REFERENCES:

- Jeffrey D. Ullman “Principles of Database Systems”, Third Edition, Galgotia Publication Pvt. Ltd.
- R. Elmasri, S.B. Navathe, “Fundamentals of Database Systems”, Fifth Edition,Pearson Education/Addison Wesley, 2008.
- .Henry F Korth, Abraham Silberschatz, S. Sudharshan, “Database System Concepts”, sixth Edition, McGraw Hill, 2011.
- C.J.Date, A.Kannan, S.Swamynathan, “An Introduction to Database Systems”, Eighth Edition, Pearson Education, 2006.
- Alex Berson and Stephen J.Smith, “Data Warehousing, Data Mining and OLAP”, Tata McGraw – Hill Edition, Thirteenth Reprint 2008.
- Jiawei Han and Micheline Kamber, “Data Mining Concepts and Techniques”, Third Edition, Elsevier, 2012.

BLUEPRINT

Code number: **CS8121**

Title of the paper: Advanced Database Management Systems

Chapter	Number of Hours	Total marks for which the question are to be asked (including bonus questions)
UNIT I	12	10
UNIT II	12	20
UNIT III	12	20
UNIT IV	12	16
UNIT V	12	14
TOTAL	60	80
Maximum marks for the paper (Excluding bonus questions) = 70		

Course Outcome

CO1	Knowledge	Have developed a good knowledge of the DBMS tools used to design Database
CO2	Understand	Have developed a very good understanding of advanced feature to modify the content of the database
CO3	Apply	Be able to implement various commands in manipulating and maintaining the Database for various entities
CO4	Analyze	Able to estimate the query based cost for an efficient and dynamic Database.
CO5	Evaluate	Able to select best tools required for the front end design along with database.
CO6	Create	Able to create complete Database and maintain without anomalies.

Course Outcomes and Course Content

Semester	II
Paper Code	**CS8221**
Paper Title	**MACHINE LEARNING WITH PYTHON**
Number of teaching hours per week	04
Total number of teaching hours per semester	60
Number of credits	04

Objective of the Paper:

This paper enables students to acquire basic knowledge in machine learning techniques and learn to apply the techniques in the area of pattern recognition and data analytics. Also this paper introduces python programming language as a machine learning tool.

UNIT 1 **12Hrs**

INTRODUCTION

Machine Learning, types of machine learning, examples. Supervised Learning: Learning class from examples, learning multiple classes, regression, model selection and generalization, Parametric Methods, parametric classification.

UNIT 2 **12Hrs**

DIMENSIONALITY REDUCTION

Introduction, subset selection, principal component analysis, factor analysis, linear discriminant analysis.

CLUSTERING

Introduction, mixture densities, k-means clustering, hierarchical clustering, Spectral clustering, choosing the number of clusters.

NON PARAMETRIC METHODS:

Introduction, non-parametric classification, distance based classification.

UNIT 3 **12Hrs**

DECISION TREES

Introduction, univariate trees, pruning, rule extraction from trees, learning rules from data.

BAYESIAN CLASSIFIER

Conditional probability, Bayes Theorem, Naïve Bayes algorithm, using numeric features with Naïve Bayes algorithm.

MULTILAYER PERCEPTRON

Introduction, training a perceptron, learning Boolean functions, multilayer perceptron,

Back propagation algorithm, training procedures.

UNIT 4 **12Hrs**

KERNEL MACHINES

Introduction, optical separating hyper plane, v-SVM, kernel tricks, vertical kernel, defining kernel, multiclass kernel machines, one-class kernel machines.

HIDDEN MARKOV MODELS

Introduction, discrete Markov processes, hidden Markov models, basic problems of HMM, evaluation problem, finding the state sequence, learning model parameters, Continuous observations, HMM with inputs, model selection with HMM.

REINFORCEMENT LEARNING

Introduction, single state case, elements of reinforcement learning, temporal difference learning, generalization, partially observed state.

UNIT 5 **12Hrs**

MACHINE LEARNING WITH PYTHON

Data framing: numpy: Ndarray, Array attributes Array creation routines, Indexing and slicing, Array Broadcasting, Array manipulation, Mathematical functions, Statistical functions, Search, sort and counting functions, Matrix Library, Linear algebra.

Pandas: Series, Data frame, Panel, Basic functionality: axes, dtype, empty, ndim, size, values, head, tail. Descriptive Statistics, Reindexing, iterations, sorting, options and Customization, Indexing and Selecting Data, Statistical Functions,Window Functions, Aggregation, Missing data, Group by, Merging, concatenation, Categorical data, I/O tools:read_csv,read_table.

Data Visualization: Matplotlib, Barplot, Histograms, Box plots, Area plot, Scatter plot, Pie chart,

Scikit: Classifiers: K-nearest, SVM, Naive base, Linear Regression. Clustering: K Means, Spectral, Hierarchical, DBSCAN, OPTICS.

Dimensionality Reduction: PCA,MDS, LDA.

SELF STUDY –6 hours of self-study will be assigned from the above units.

References

- E. Alpaydin, Introduction to Machine Learning. 2nd MIT Press, 2009.
- K. P. Murphy, Machine Learning: A Probabilistic Perspective. MIT Press, 2012.
- P. Harrington, Machine Learning in Action. Manning Publications, 2012
- C. M. Bishop, Pattern Recognition and Machine Learning. Springer, 2011.
- Andreas C. Muller, Sarah Guido, Introduction to Machine Learning with Python

BLUEPRINT

Code number: **CS8221**

Title of the paper: **MACHINE LEARNING WITH PYTHON**

Chapter	Number of Hours	Total marks for which the questions are to be asked (including bonus questions)
Unit I	12	14
Unit II	12	20
Unit III	12	20
Unit IV	12	16
Unit V	12	10
TOTAL	**60**	**80**
Maximum marks for the paper (Excluding bonus question)= 70		

Practical I

MCS 2P1-Machine Learning Using Python

(11 sessions 6 hr/week)

1. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.
2. Demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
3. Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets.
4. Implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data set
5. Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Calculate the accuracy, precision, and recall for your data set.

6. Apply K-Means clustering algorithm to cluster a set of data stored in a .CSV file.
7. Implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions.
8. Implement the parametric Linear Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.

Course Outcomes: At the end of the course, the student should

CO1	Knowledge	Have developed a good knowledge of basic principles of machine learning techniques and have developed a good knowledge of machine learning capabilities of python.
CO2	Understand	Have developed a very good understanding of types of machine learning techniques like supervised and un supervised learning.
CO2	Apply	Be able to use various machine learning models.
CO3	Analyze	Be able to compare various machine learning models and select a suitable model for a given problem.
CO4	Evaluate	Be able to evaluate various models in python and select the appropriate one for a given real time problem
CO5	Create	Be able to design and build small machine learning applications which can be used to solve various real time problems.

Course Outcomes and Course Content

Semester	II
Paper Code	**CS8321**
Paper Title	**Advanced Web Technologies.**
Number of Teaching hours per week	04
Total number of teaching hours per semester	60
Number of credits	04

Objectives

This paper provides an insight to develop a website using essential tools of Angular and an integration of ASP .NET for interactive web pages. This paper also focuses on back end support in building queries and introduction of Firebase.

UNIT I

Basic elements of HTML and JavaScript

a. HTML **3 Hrs**

Basic syntax, Standard structure, Basic text markup, Images, Hyper Links. Lists, Tables, Forms, Frames (attributes for all HTML elements).

b. **Cascading Style Sheets** 4 **Hrs**

Introduction, Style syntax ,Types of style sheets, , Selector forms, Property value forms, Font properties, List properties, Color, Alignment of text, The box model, Background images, The and tags.

c. **The Basics of JavaScript** 5 **Hrs**

Overview of JavaScript, Object orientation and JavaScript, Syntactic characteristics, data types, operators and expressions, Screen output and keyboard input, Control statements, Object creation and modification, Arrays, Strings, Functions(built-in and user defined), Constructors, DOM,Events and Event Handling.

UNIT II

ES6, TypeScript, Angular-CLI and Angular Components
a. ES6 1 **Hrs**

Module system-Classes-Variable declaration-Arrow Functions-Template Strings.

b. Type Script 2 **Hrs**

Type safety, inference and intellisense-Interfaces-Decorators.

c. Angular 3 **Hrs**

CLI & project structure-Creating a new project-Project settings, bootstrapping-Building and serving-Component-based architecture-Angular building blocks overview-Generating project elements-Root

d. Angular Components 3 **Hrs**

Angular Components-Component definition-Component types-Template syntax-Data, property and event

Binding-Using directives and pipes-Inputs-Outputs (events)-Component style

e. Advanced Components 3 **Hrs**

Data projection, building a wrapper component-Querying view and children

Dynamic components

UNIT III

Providers, Dependency Injection, Observables 4 **Hrs**

a. Providers and Dependency Injection-Understanding the role of the Provider-Understanding the injector tree-Creating and using a class provider (service)-Other provider strategies-Configuring providers.

b**. Observables and RxJS** 2 **Hrs**

Subscription-RxJS Operators-Creating Subjects and Observables

c. **HttpClient** 3 **Hrs**

Http requests (GET, PUT, and POST)

Configuring headers-Interceptors-Progress events

d. Angular Router 3 **Hrs**

Setting up the router-Navigation-Child routes-Routing params-Lazy loading-Guards and hooks

UNIT IV

Angular Modules, directives and Pipes

a. Angular Modules and Directives 3 **Hrs**

Root Module vs. Feature Module-Module definition-Module configuration

Types of directives-Built-in directives-Writing your own directives

b. Pipes-Sync and async Pipes 3 **Hrs**

Built-in Pipes-Writing your own pipes, Angular Forms

c. Template-driven forms-Reactive forms 3 **Hrs**

Form Builder, Form validation, Custom validators and Async validators

d. State management with ngRx 3 **Hrs**

Understanding Redux architecture-ngRx Store-Actions & Reducers-Middleware-Effects and Facades-Action Splitters

UNIT V

ASP.NET Core and API Fundamentals

a. Building an API with ASP.NET Core 4 **Hrs**

Introduction to REST and HTTP-Designing the API & URI

Using Status Codes-Using Get for Collections

b. Using Query Strings 4 **Hrs**

Modifying Data-Implementing POST, PUT & DELETE. Integration of front end and back end.

c. Fire Base 4 **Hrs**

Overview, Environment Setup, Data, Arrays, write Data, Write List Data, Write Transactional Data, Read Data, Event Types, Detaching Callbacks and Queries.

NOTE: 6 hours of self- study will be assigned from the above units.

REFERENCES

- HTML and CSS: Design and Build Websites
- Mastering HTML, CSS & Javascript by Web Publishing
- Angular — The Complete Guide by Maximilian Schwarzmüller.
- The Ng-book — The Complete Book on Angular by Nate Murray, Felipe Coury, Ari Lerner
- The Complete Angular Course: Beginner to Advanced by Mosh Hamedani.
- "Professional ASP.NET MVC 5 (WROX)" by Jon Galloway and Brad Wilson
- "ASP.NET: The Complete Reference" by Matthew Macdonald
- Firebase by Sam Sisavath

BLUEPRINT

Code number: **CS8321**

Title of the paper: **Advanced Web Technologies.**

Chapter	Number of Hours	Total marks for which the question are to be asked (including bonus questions)
UNIT I	12	8
UNIT II	12	20
UNIT III	12	20
UNIT IV	12	16
UNIT V	12	16
TOTAL	60	80
Maximum marks for the paper (Excluding bonus questions) = 70		

Practical II

MCS2P2- Advanced Web Technologies &ADBMS Lab
(11 sessions 6hrs/week)

Program List

1. Design a web page using the following elements for a company
    a. Basic tags, text formatting ,image, anchor, lists, table, forms tags
    b. Create a home page using CSS concepts to add a menu and various text, image and color properties.
    c. Design two forms that includes all the Java Script Objects(text box, text area, button, radio button, check box)
2. Demonstrate Angular module and components.
3. Demonstrate Angular components.
4. Demonstrate Angular Providers.
5. Demonstrate Angular Dependency Injection.
6. Demonstrate Angular router.
7. Demonstrate Angular Modules.
8. Demonstrate Angular directives and Pipes.
9. Demonstrate Angular forms.
10. Demonstrate Angular State management with ngRx.
11. Demonstrate front end and backend.
12. Database Customization using SQL
13. Creating Databases/Table spaces /constrains/keys
14. Create Objects
15. Moving Data
16. Recovery
17. Locking
18. Preparing Applications for Execution using a front end tool
19. Application Performance Tool

Course Outcome: At the end of the course the student should

CO1	Knowledge	Have developed a good knowledge of the tools (Angular) used to design web page with a backend.
CO2	Understand	Have developed a very good understanding of advanced techniques to develop and test the web pages.
CO3	Apply	Be able to integrate web based applications with suitable database applications.
CO4	Analyze	Able to test and debug the codes in developing effective web sites.
CO5	Evaluate	Able to compare and implement better codes to reduce the complexity of the web design process.
CO6	Create	Able to create and host the web site with domain knowledge.

Course Outcomes and Course Content

Semester	II
Paper Code	**CS8421**
Paper Title	**Principles of Compiler design**
Number of teaching hours per week	04
Total number of teaching hours per semester	60
Number of credits	04

Objective of the Paper:

Compiler Design will teach students the fundamental concepts and techniques used for building a simple compiler,To understand the phases of complier, creation and execution of parse tree with symbol tables, errors encountered by the complier and code optimization. The discussion will also include the examination of intermediate code states, machine code optimization techniques and support for advanced language features

UNIT-I

Introduction to compilers **12 Hours**

Analysis of source program – Phase of compiler – Cousins of compilers – Simple one pass compiler: overview – Syntax definition Lexical analysis: removal of white space and comments – Constants – Recognizing identifiers and keywords – Lexical analysis – Role of a lexical analyzer – Input buffering –Specification of tokens – Recognition tokens-Introduction to LEX .

UNIT-2

Symbol Tables **12 Hours**

Symbol table entries – List data structures for symbol table – - Hash tables – Representation of scope information – Syntax Analysis: Role of parser – Context free grammar – Writing a grammar – Top down parsing: Brute force method, Recursive Descent, LL(1) parser – Simple bottom up parsing:– Shift reducing parsing, Operator precedence ,LR parsers.

UNIT- 3

Syntax Analysis **12 Hours**

Syntax directed definition: Construction of syntax trees – Bottom up evaluation of S- Attributed definition – L-Attributed definitions – Top down translation - Type checking: Type systems – Specifications of simple type checker. Error Recovery: Error Detection & Recovery, Ad-Hoc and Systematic Methods, Error Handling and Recovery in Syntax Analyzer-YACC

UNIT-4

Run time Environment **12 Hours**

Run-time environment: Source language issues – Storage organizations – Storage location strategies - Intermediate code generation: Intermediate languages – Declarations – Assignment statements Code Optimization: Code Optimization Global Data Flow Analysis, A Few Selected Optimizations like Command Sub Expression Removal, Loop Invariant Code Motion, Strength Reduction .

UNIT-5

Code Generation **12 Hours**

Issues in the Design of a Code Generator, The Target Machine, Run-Time Storage Management, Basic Blocks and Flow Graphs, Next-Use Information, A Simple Code Generator, Register Allocation and Assignment, The DAG Representation of Basic Blocks, Peephole Optimization, Generating Code from DAGs, Dynamic Programming Code-Generation Algorithm, Code Generator Generators.

REFERENCES:

- Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman, “Compilers :principles,Techniques, and Tools, Pearson Education Asia.
- Dhamdhere D.M., “Compiler Construction: Theory andPractice”,McMillan India Ltd.
- Holub Allen, “Compiler Design in C”, Prentice Hall of India.(2ndRevised Edition)

BLUEPRINT

Code number: **CS8421**

Title of the paper: **Network and Cyber Security**

Chapter	Number of Hours	Total marks for which the questions are to be asked (including bonus questions)
Unit I	12	14
Unit II	12	20
Unit III	12	20
Unit IV	12	16
Unit V	12	10
TOTAL	**60**	**80**
Maximum marks for the paper (Excluding bonus question)= 70		

Course Outcomes: At the end of the course, the student should

CO1	Knowledge	Have developed a good knowledge about the role of complier to translate the source code to objectcode.
CO2	Understand	Have developed a very good understanding on the six phases ofcomplier and increase the efficiency of the compiler in understanding Analysis and synthesis part.
CO2	Apply	Be able toWrite a scanner, parser, and semantic analyser without the aid of automatic generators
CO3	Analyse	Be able to Specify and analyse the lexical, syntactic and semantic structures of advanced language
CO4	Evaluate	Be able to Design the structures and support required for compiling advanced language features.
CO5	Create	Be able to create a compiler using the tools YACC and LEX

Course Outcomes and Course Content

Semester	II
Paper Code	**CS8521**
Paper Title	**SOFTWARE PROJECT MANAGEMENT**
Number of teaching hours per week	04
Total number of teaching hours per semester	60
Number of credits	04

Objective of the Paper:

This paper introduces the students to the fundamentals of software project management and to give an insight into the project planning phase, selection of an appropriate project approach and an in-depth study of software project models. This paper will get the students to understand the concepts of effort estimation, activity planning, risk management in software projects. The paper formally introduces the students to the concepts of resource management, role of quality in software and build people management skills.

UNIT 1 **12 Hrs**

INTRODUCTION TO SOFTWARE PROJECT MANAGEMENT

Introduction,Software project versus other types of projects, contract management and technical project management, activities covered by Software Project Management, plans methods and Methodologies, Categorizing Software Projects, setting objectives stakeholders, business case, requirement specification, Management control.

Project planning:Introduction and various steps in project planning.

UNIT 2 **12 Hrs**

SELECTION OF APPROPRIATE PROJECT APPROACH

Choosing technologies, technical plan contents list, choice of process models, (structure versus speed of delivery), waterfall model, V-process model, spiral model, software prototyping, other ways of categorizing prototypes, controlling changes during prototyping, incremental delivery,(dynamic systems development method(Agile model, scrum, role of scrum master), extreme programming, Managing iterative processes), selecting more appropriate process model.

UNIT 3 **12 Hrs**

SOFTWARE EFFORT ESTIMATION AND ACTIVITY PLANNING

Estimation Techniques top-down estimation, bottom-up estimation, Albrecht function point analysis, expert judgment, staffing pattern.

Activity planning: Objectives of plan, project schedules, projects and activities, sequencing and scheduling the activities, network planning models, formulating a network model, adding time dimension, forward pass, backward pass, identifying critical path, activity float, shortening project duration, identifying critical activities, activity-on-arrow networks.

UNIT 4 **12 Hrs**

RISK MANAGEMENT

Nature of risk, types of risk, managing risk, hazard identification, hazard analysis, risk planning and control, evaluating risks to the schedule.

RESOURCE ALLOCATION: Nature of resources, identifying resource requirements, scheduling resources, creating critical paths, counting the cost, publishing resource schedule.

UNIT 5 **12 Hrs**

MANAGING PEOPLE AND ORGANIZING TEAMS

Understanding behavior, organizational behavior: a background, selecting the right person for the job, instruction in the best methods, motivation, working in groups, becoming a team, decision making, leadership, organizational structures, stress, health and safety.

SOFTWARE QUALITY

Place of software quality in project planning, importance of project quality, defining software quality, ISO 9126, practical software quality measures, techniques to help enhance software quality, quality plans.

Self study: COCOMO Model, various process models, case study on project planning and any one of the ISO standards.

REFERENCES

- Hughes, Bob and Cotterell, Mike, Software project Management, 4th Edition, TMH.
- Kathy Schwalbe, Information Technology Project Management, VikasPublishing House.
- Kieron Conway, Software Project Management – From Concept toDeployment,
- Kelkar S. A,Information Technology Project Management, A concise study,PHI, 2005

BLUEPRINT

Code number: **CS8521**

Title of the paper: **SOFTWARE PROJECT MANAGEMENT**

Chapter	Number of Hours	Total marks for which the questions are to be asked (including bonus questions)
Unit I	12	14
Unit II	12	20
Unit III	12	20
Unit IV	12	16
Unit V	12	10
TOTAL	**60**	**80**
Maximum marks for the paper (Excluding bonus question)= 70		

CO1	Knowledge	Have developed a good knowledge of the fundamentals of the discipline of Software Project Management and the role of Managers in managing projects
CO2	Understand	Have developed a very good understanding of the different process models, effort estimation techniques, risk analysis techniques, quality assessment methods, managing teams and understanding team structures
CO2	Apply	Be able to apply the policies for better management of software projects.
CO3	Analyze	Be able to analyze various techniques and methods of management of all the aspects of a software project and select the best approach.
CO4	Evaluate	Be able to critique the various techniques of software project management
CO5	Create	To be able to work in a group as a team leader or active team member in an IT project.