SHIVAJI UNIVERSITY KOLHAPUR

REVISED SYLLABUS AND STRUCTURE

THIRD YEAR (B. Tech.) (CBCS)

Computer Science and Engineering

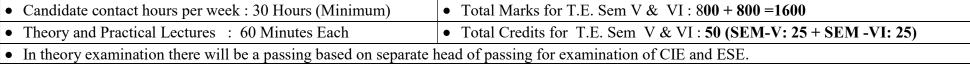To be introduced from the academic year 2020-21

(From June 2020) onwards

(Subject to the modifications will be made from time to time)

		TEACHING SCHEME									EXAMINATION SCHEME										
		THEORY			TUTORIAL			PRACTICAL			THEORY					PRACTICAL			TERMWORK		
Sr. No.	Course Subject / Title	Credits	N0. Of Lectures	Hours	Credits	No. of Hours	Hours	Credits	No. of Hours	Hours	Hours	mode	marks	Total Marks	MIN.	Hours	MAX	MIN.	Hours	MAX	MIN.
1	PCC-CS501 Information Security	3	3	3				1	2	2		CIE	30	100	40	AS PER BOS GUIDELINES			AS PER BOS GUIDELINES	50	20
												ESE	70								
2	PCC- CS502 System Programming	3	3	3				1	2	2		CIE	30	100	40		25	10		50	20
												ESE	70								
3	PCC- CS503 Object-Oriented Modeling & Design	3	3	3								CIE	30	100	40						
												ESE	70								
4	PCC- CS504 Computer Algorithms	4	4	4	1	1	1					CIE	30	100	40					25	10
												ESE	70								
5	OEC- CS505	3	3	3								CIE	30	100	40						
												ESE	70								
6	PCC- CS506 Java Programming	3	3	3				2	4	4							50	20		50	20
7	HM- CS507 Business English				1	1				2							25	10		25	10
	Total (SEM –V)	19	19	19	2	2	2	4	8	8				500			100			200	

THIRD YEAR COMPUTER SCIENCE AND ENGINEERING - CBCS PATTERN

SEMESTER - VI

Sr. No.	Course Subject / Title	TEACHING SCHEME									EXAMINATION SCHEME									
		THEORY			TUTORIAL			PRACTICAL			THEORY				PRACTICAL			TERMWORK		
		Credits	N0. Of Lectures	Hours	Credits	No. of Hours	Hours	Credits	No. of Hours	Hours	Hours	mode	marks	Total Marks	MIN.	Hours	MAX	MIN.	Hours	MAX	MIN.
1	PCC-CS601 Compiler Construction	3	3	3				1	2	2		CIE	30	100	40	AS PER BOS GUIDELINES			AS PER BOS GUIDELINES	25	10
												ESE	70								
2	PCC- CS602 Operating System	4	4	4				1	2	2		CIE	30	100	40					25	10
												ESE	70								
3	PCC- CS603 Database Engineering	4	4	4				1	2	2		CIE	30	100	40		50	20		25	10
												ESE	70								
4	PCC- CS604 Machine Learning	3	3	3	1	1	1					CIE	30	100	40						
												ESE	70								
5	OEC- CS605	3	3	3								CIE	30	100	40					25	10
												ESE	70								
6	PCC- CS606 C# Programming	2	2	2				1	2	2							50	20		25	10
7	PW- CS607 Domain Specific Mini Project							1	2	2							50	20		25	10
	Total (SEM –VI)	*19*	*19*	*19*	*1*	*1*	*1*	*5*	*10*	*10*				*500*			*150*			*150*	
	Total	**38**	**38**	**38**	**3**	**3**	**3**	**9**	**20**	**20**				**1000**			**250**			**350**	

CIE- Continuous Internal Evaluation

ESE – End Semester Examination

● Candidate contact hours per week : 30 Hours (Minimum)	● Total Marks for T.E. Sem V & VI : 8**00 + 800 =1600**
● Theory and Practical Lectures : 60 Minutes Each	● Total Credits for T.E. Sem V & VI : **50 (SEM-V: 25 + SEM -VI: 25)**
● In theory examination there will be a passing based on separate head of passing for examination of CIE and ESE.	
● There shall be separate passing for theory and practical (term work) courses.	

Note:

1. **PCC-CS:** Professional Core Course – Computer Science and Engineering are compulsory.
2. **HM-CS:** Humanities and Management- Computer Science and Engineering are compulsory.
3. **PW-CS:** Domain Specific Mini Project -– Computer Science and Engineering are compulsory.
4. **OEC-CS: Open Elective Course** – Computer Science and Engineering are compulsory.

T. Y. B.Tech (Computer Science and Engineering) Sem – V

PCC-CS501 Information Security

TEACHING SCHEME	EXAMINATION SCHEME
Theory : 03 Hrs / Week	**Term work:** 50 Marks
Tutorial : --	**Theory** : ESE 70 Marks CIE 30 Marks
Practical: 02 Hrs / Week	**Practical** : --

Prerequisite: Computer algorithms **,** Computer Networks

Course Objectives:
1. Understand basics of cryptography, and some key encryption techniques.
2. Understand modern cryptosystems.
3. Understand network and Web security protocols.

Course Outcomes:
Upon successful completion of this course, the student will be able to –
1. Describe basic terminology in cryptography, and classical cryptosystems.
2. Explain modern cryptosystems.
3. Explain security policies such as authentication, integrity and confidentiality.
4. Explain network and Web security protocols.

Unit No.	Unit Name and Contents	No. of Lectures
1	Overview and Classical Encryption Techniques Overview: Computer Security Concepts, The OSI Security Architecture, Security Attacks, Security Services, Security Mechanisms, A Model for Network Security Classical Encryption Techniques: Symmetric Cipher Model, Substitution Techniques, Transposition Techniques, Rotor Machines	6
2	Block Ciphers and Advanced Encryption Standard Block Cipher Principles, The Data Encryption Standard (DES), A DES Example, The Strength of DES, Differential and Linear Cryptanalysis, Block Cipher Design Principles Finite Field Arithmetic, AES Structure, AES Transformation Functions, AES Key Expansion, An AES Example, AES Implementation	6
3	Public Key Cryptography –Principles of Public-Key Cryptosystems, The RSA Algorithm, Diffie-Hellman Key Exchange, ElGamal Cryptosystem	6
4	Cryptographic Data Integrity Algorithms Cryptographic Hash Functions: Applications of Cryptographic Hash Functions, Two Simple Hash Functions, Requirements and Security, Hash Functions Based on Cipher Block Chaining, Secure Hash Algorithm (SHA), SHA-3 Message Authentication Code: Message Authentication Requirements, Message Authentication Functions, Requirements for Message Authentication Codes, Security of MACs, MACs Based on Hash	9

	Functions: HMAC, MACs Based on Block Ciphers: DAA and CMAC, Authenticated Encryption: CCM and GCM Digital Signatures: Elgamal Digital Signature Scheme, Schnorr Digital Signature Scheme, NIST Digital Signature Algorithm	
5	Mutual Trust and Key Management Key Management and Distribution: Symmetric Key Distribution Using Symmetric Encryption, Symmetric Key Distribution Using Asymmetric Encryption, Distribution of Public Keys, X.509 Certificates, Public-Key Infrastructure	**6**
6	Network and Internet Security Transport-Level Security: Web Security Considerations, Secure Sockets Layer, Transport Layer Security, HTTPS Electronic Mail Security: Pretty Good Privacy, S/MIME, Domain Keys Identified Mail IP Security: IP Security Overview, IP Security Policy, Encapsulating Security Payload, Combining Security Associations, Internet Key Exchange, Cryptographic Suites	**6**

Text Books:

1. Williams Stallings – Cryptography and Network security principles and practices, Pearson Education (LPE), Sixth Edition

Reference Books:

1. Cryptography and network security – Atul Kahate (TMGH)
2. Handbook of Applied Cryptography - Menezes, A. J., P. C. Van Oorschot, and S. A. Vanstone

PCC-CS502 – System Programming

TEACHING SCHEME	EXAMINATION SCHEME
Theory : 3 Hrs/Week	**Term work:** 50Marks
Tutorial : --	**Theory** : ESE 70 Marks CIE 30 Marks
Practical: 2 Hrs/Week	**Practical** : 25 Marks

Prerequisite: Microprocessor & Its programming, data structures

Course Objectives:

1. To understand fundamentals of languages and processing
2. To make students to learn design of grammars, assemblers and compilers
3. To provide hands on experience to the students on simulation of linkers, loaders and software tools for UIs

Course Outcomes:

Upon successful completion of this course, the student will be able to –

1. Identify the role of system programs and application programs.
2. Understand the basics of system programs like editors, compiler, assembler, linker, loader, interpreter and debugger.
3. Describe the various concepts of assemblers and macro - processors.
4. Understand the various phases of compiler and compare its working with assembler.
5. Understand how linker and loader create an executable program from an object module created by assembler and compiler.
6. Create graphical user interfaces for basic programs and learn about terminal input/output.

Unit No.	Unit Name and Contents	No. of Lectures
1	**Language Processors**: Introduction, Language processing activities, Fundamentals of language processing, Fundamentals of language, Specification, Language Processing Development Tools: LEX and YAC	7
2	**Assemblers:** Elements of assembly language programming, A simple assembly scheme, pass structure of assemblers, Design of a two pass assemble	5
3	**Macros and Macro Processors:** Macro definition and call, Macro expansion, Nested macro calls, Advanced macro facilities, Design of macro pre-processor: Design Overview, Data structure of Macro pre-processor with and Example	7
4	**Compilers and Interpreters:**	7

	Aspects of compilation, Memory allocation: Static and Dynamic memory allocation, Memory Allocation in block Structure language. Compilation of expressions, Code optimization: Local and Global optimization and Optimization technique, Interpreters	
5	**Linkers:** Introduction, Relocation and linking concepts, Self- relocating programs, linking for overlays, Loaders	**6**
6	**Software tools:** Introduction, Software tools for program development, Editors, Debug Monitors, Programming Environments, and User Interface	**6**

Term Work: Minimum of 5 experiments on LEX and 5 case study each on Assembler, Complier, Macro Preprocessor, Linker and Loaders. **F**ollowing list can be considered:

1. Study of Language Processor Development tools using LEX and YAAC.
2. Design and Implement Pass-I of an assembler.
3. Design and implement Pass-II of an assembler.
4. Implementation of Macro Expansion.
5. Study the design of Macro-Assembler.
6. Implement Parameter passing Mechanisms.
7. Design and implement an Interpreter.
8. Implement program relocation and Linker Concepts.
9. Study of Loaders.
10. Study of Software tool and its types.

Orals can be conducted over the syllabus contents and Termwork assignments

Text Books:

1. System Programming and operating systems, D.M.Dhamdhere, 2nd Edition, (TMGH), All Units
2. Lex &Yacc Publisher: Doug Brown, John Levine, Tony Mason2nd EditionO'Reilly MediaFor Practical

PCC- CS503 Object-Oriented Modeling & Design

TEACHING SCHEME	EXAMINATION SCHEME
Theory : 03 Hrs / Week	**Term work:** --
Tutorial : --	**Theory** : ESE 70 Marks CIE 30 Marks
Practical: --	**Practical** : --

Prerequisite: Software Engineering Concepts, Object Oriented Concepts

Course Objectives:

1. To Understand the Object Based View of the System

2. To design Problems using Object Oriented Analysis and Design Techniques

3. To Understand UML notations and compare with OMT

4. To inculcate necessary skills to handle complexities in Software Design

Course Outcomes:

Upon successful completion of this course, the student will be able to –

1. Analyze and model software systems

2. Construct Object Oriented view of the system

3. Able to design a Software system using OMT and UML design techniques.

Unit No.	Unit Name and Contents	No. of Lectures
1	**Introduction:** Object oriented themes, modeling as a design technique. **Object Modeling:** Object, classes, Link & association, advanced link & Association concepts, generalization & Inheritance, grouping constructs, aggregation, abstract classes, generalization as extension & restriction, multiple inheritance, metadata, candidate key & constraints.	8
2	**Dynamic & Functional Modeling:** **Dynamic modeling**: Events & states, operations, nested state diagrams, concurrency, advanced dynamic modeling concepts & simple dynamic model, relation of object dynamic models. **Functional Modeling**: functional model, data flow diagrams, specifying operations, constriction, a simple functional model, relation of functional to object & dynamic model.	7

3	**Design Methodology:** OMT methodology, Impact of an object oriented approach, analysis, system design with examples, combining models, design algorithms, design optimization, implementation of controls, design association & physical packaging.	5
4	**Introducing the UML:** An overview of the UML, Conceptual Model of UML, Architecture of UML **Structure modeling Using UML:** Classes, Relationship, Diagrams, Class Diagrams.	7
5	**Behavioral Modeling:** Interactions, Use Cases, Use Case Diagram, Interaction diagrams, Activity diagrams, Events & Signals, State Machines, Process & Threads, Time & Space, State chart diagrams.	6
6	**Architectural Modeling:** Components, Deployment, Collaboration, Patterns & frameworks, component diagrams, Deployment diagrams.	6

Text Books:

1. Object-orientated Modeling & Design: (Unit 1 to 3), James Rambaugh, Michael Blaha, William Premerlani, Frederick Eddy, William Lorensen, PHI Unit 1 to 3
2. The Unified Modeling Language User Guide, Grady Booch, James Rambaugh, Lvar Jacobson, Unit 4 to 6

Reference Book:

1. Object oriented analysis & design using UML,H. Srimathi, H. Sriram, A.Krishnamoorthy SCITECH PUBLICATION 2nd Edition
2. Object Oriented analysis& Design Andrew High TMG
3. Practical Object Oriented Design with UML Mark Priestley
4. Threat first Object oriented analysis & design, Breet Mclaughline, Garry Police & Devide West OREILLY

PCC- CS504 Computer Algorithms

TEACHING SCHEME	EXAMINATION SCHEME
Theory : 04 Hrs./Week	**Term work:** 25 Marks
Tutorial :01 Hrs/ Week	**Theory** : ESE 70 Marks CIE 30 Marks
Practical: --	**Practical** : --

Prerequisite: Problem analysis and program design in C, Object Oriented Design using C++ , Data Structure

Course Objectives:

1. To understand fundamentals of algorithms.
2. To learn different algorithm design paradigms.
3. To perform performance analysis of algorithms.
4. To understand different data structures and operations on them.

Course Outcomes:

Upon successful completion of this course, the student will be able to –

1. Explain fundamentals of algorithms.
2. Apply Divide and Conquer and Greedy Approach to tackle problems.
3. Solve the problems using dynamic programming and backtracking.
4. Analyze performance of algorithms using asymptotic analysis.
5. List different data structures and operations on them.

Unit No.	Unit Name and Contents	No. of Lectures
1	**Introduction to Algorithms**– Introduction, Characteristics of algorithms, Pseudo code Conventions, Recursive Algorithms, Performance analysis, Asymptotic notations (O, Ω, Θ), Performance measurement, Randomized Algorithms, Recurrence relations.	6
2	**Divide and Conquer**– General method, Binary search, finding the maximum and minimum, Merge sort, Quick sort, Selection, Convex Hull.	6
3	**The Greedy method** – General method, , Knapsack problem, Job sequencing with deadlines, minimum-cost spanning trees – Prim's and Kruskal's Algorithms, Optimal storage on tapes , Optimal merge patterns, Huffman codes Single source shortest paths.	7
4	**Dynamic Programming** – General method, Multistage graphs, All pair shortest paths, Optimal binary search trees, 0/1 knapsack, Reliability design, Traveling Sales person problem, Flow shop scheduling.	6

5	**Backtracking** – General method, n-queen problem, Permutation tree, Sum of subsets, Knapsack Problem, Hamiltonian Cycle, and Graph Coloring.	6
6	**NP Hard and NP Complete Problems** – Basic Concepts, P, NP, NP Complete, NP Hard, Cook's Theorem, NP Hard Graph problems, NP Hard Scheduling problems, NP Hard Code Generation Problems.	7

Text Books:

1. "Fundamentals of Computer Algorithms", Ellis Horowitz, SartajSahni, SanguthevarRajasekaran, 2nd Edition, Universities Press.
2. "Introduction to Algorithms", Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, 3rd Edition, The MIT Press.

Reference Books:

1. "The Design and Analysis of Computer Algorithms", A. Aho, J. Hopcroft and J. Ullman, 1st Edition, Addison-Wesley
2. "Introduction to The Design and Analysis of Algorithms", Anany Levitin, 3rd Edition, Pearson.

OEC- CS605-Computer Graphics

TEACHING SCHEME	EXAMINATION SCHEME
Theory : 03 Hrs / Week	**Term work:** 50 Marks
Tutorial : --	**Theory** : ESE 70 Marks CIE 30 Marks
Practical: --	**Practical** : --

Prerequisite: Data Structures, Engineering Mathematics, Engineeringdrawing
Course Objectives:

1. To understand basics of computer graphics and different display devices.
2. To implement 2D and 3Dtransformations in computer graphics.
3. To expose students to the various raster scan algorithms.
4. To make the students aware of generation of curves and surfaces.
5. To provide knowledge to the students about basics of Illumination models, surface rendering methods and Multimedia.

Course Outcomes:

Upon successful completion of this course, the student will be able to –

1. Express basic ideas of computer graphics and different Display Devices.
2. Demonstrate 2D and 3D transformations.
3. Make use of various known raster scan algorithms to perform different operations on object.
4. Demonstrate different types of curves in computer graphics.
5. Explain basics of Illumination models, surface rendering methods and Multimedia

Unit No.	Unit Name and Contents	No. of Lectures
1	**Introduction to Computer Graphics:**Introduction to raster scan display, frame buffer, graphics primitives, color models, display devices: LED display, AMOLED display, HD display3D display, Virtual Reality display.	4
2	**Transformations:**Basic 2D & 3D transformations - Translation, Scaling, Rotation, Reflection, Shearing, Multiple Transformations, Rotation about an axis parallel to a coordinate axis, rotation about an arbitrary axis in space, Affine and Perspective Geometry, Orthographic projections and Axonometric projections.	**4**
3	**Raster Scan Graphics:** Bresenham's line and circle drawing algorithms, Scan Conversion techniques: RLE, Frame Buffer, Scan converting polygons: Edge fill and Seed fill algorithms, Anti-aliasing and Halftoning, Clipping:- Windowing and View-porting, Introduction to clipping, Point clipping, and line clipping: Sutherland - Cohen line clipping algorithm	**6**
4	**Curves and Surface:** Curve Representation, Non-parametric and parametric curves, representation of space curves, Cubic Spline, Parabolic Blended curves,	**6**

	Bezier curves and B-spline curves, Z- buffer, Warnock algorithm.	
5	**Illumination models and surface rendering methods**. Light sources, Basic illumination models, Displaying light intensities, Halftone patterns and Dithering Techniques, Polygon Rendering methods, Ray tracing methods, Radiosity lighting	5
6	**Computer Animation &Virtual Reality :**Introduction, key-frame animation ,construction of animation sequence, Motion control methods, procedural animation. Design of VR systems, Important factors in VR system, Types of VR systems, Input and output devices for VR system.	7

Text Books:

1. Mathematical elements for Computer Graphics - David F. Rogers, J. Alan Adams (MGH Int.) (For Units 1, 2)
2. Procedural elements for Computer Graphics - David F. Rogers, J. Alan Adams (MGH Int.) (For Units 3,4)
3. Computer Graphics C Version second edition –Donald D. Hearn, M. Pauline Baker (Pearson) (For Unit 5).
4. Computer Graphics with Virtual Reality Systems, Rajesh K. Maurya, Wiley India Publications (For unit 6)

Reference Books:

1. Donald Hearn and M.Pauline Baker, "Computer Graphics C Version", Pearson Education
2. Prabat K Andleigh and Kiran Thakrar, "Multimedia Systems and Design", PHI, 2003
3. Principles of Computer Graphics Theory and Practice Using OpenGL and Maya, Shalini Govil-Pai, (Springer).
4. Computer Graphics (second Edition) - Zhigang Xiang & Roy Plastock (Schaum's Outline Series, TMGH).
5. Computer Graphics Using OpenGL F.S. Hill Jr. Stephen M. Kelley, (Pearson Education).

PCC- CS506 Java Programming

TEACHING SCHEME	EXAMINATION SCHEME
Theory :3Hrs/Week	**Term work: 5**0 Marks
Tutorial : --	**Theory :--**
Practical: 4 Hrs/Week	**Practical : 5**0 Marks

Prerequisite: Programming inC,C++, HTML
Course Objectives:
1. To Introduce the concept of object oriented programming using java
2. To learn how to implement reliable and secure application using exception handling and package concept.
3. To write program to perform file operations
4. To understand how to design components with java Swing API and present mechanism of multithreading
5. To familiarize database connectivity through JDBC and learn the collection framework
6. To explore the concept of networking and web programming using java servlet and jsp

Course Outcomes:
Upon successful completion of this course, the student will be able to –
1. Articulate the principle of object oriented problem solving & programming**.**
2. Illustrate code reusability, security and abstraction using inheritance, package and interface
3. Develop reliable and user friendly applications using exception handling and file handling
4. Create desktop apps using SWING and event handling and also illustrate multithreading concepts
5. Use JDBC & collection framework.
6. Apply network programming concept & develop web applications using servlet and jsp.

Unit No.	Unit Name and Contents	No. of Lectures
1	**Fundamental Programming in Java:** The Java Buzzwords, The Java Programming Environment- JVM, JIT Compiler, Byte Code Concept, HotSpot, A Simple Java Program, Source File Declaration Rules, Comments, Data Types, Variables, Operators, Strings, Input and Output, Control Flow, Big Numbers, Arrays-Jagged Array. **Objects and Classes:** Object-Oriented Programming Concepts, Declaring Classes, Declaring Member Variables, Defining Methods, Constructor, Passing Information to a Method or a Constructor, Creating and using objects, Controlling Access to Class Members, Static Fields and Methods, this keyword, Object Cloning, Class Design Hints,	5
2	**Inheritance, Interface and Packaging:** Inheritance: Definition, Superclasses, and Subclasses, Overriding and Hiding Methods,Polymorphism, Inheritance Hierarchies, Super keyword, Final Classes	7

	and Methods, Abstract Classes and Methods, casting, Design Hints for Inheritance, Nested classes & Inner Classes, finalization and garbage collection. Interfaces: Defining an Interface, Implementing an Interface, Using an Interface as a Type, Evolving Interfaces, and Default Methods. Packages: Class importing, Creating a Package, Naming a Package, Using Package Members, Managing Source and Class Files. Developing and deploying (executable) Jar File.	
3	**Exception and I/O Streams:** Exception: Definition, Dealing with Errors, The Classification of Exceptions, Declaring Checked Exceptions, Throw an Exception, Creating Exception Classes, Catching Exceptions, Catching Multiple Exceptions, Re-throwing and Chaining Exceptions, finally clause, Advantages of Exceptions, Tips for Using Exceptions. I/O Streams: Byte Stream – InputStream, OutputStream, DataInputStream, DataOutputStream,FileInputStream, FileOutputStream, Character Streams, BufferedStream, Scanner, File,RandomAccesFile.	**4**
4	**Graphical User Interfaces using Swing and Multithreading** Introduction to the Swing, Swing features, Swing Top Level Containers-Creating a Frame, Positioning a Frame, Displaying Information in a Panel, The Model-View-Controller Design Pattern, The JComponent Class. Layout Management: Introduction to Layout Management, APIs for Border Layout, Flow Layout, Grid Layout Event Handling: Basics of Event Handling, The AWT Event Hierarchy, Semantic and Low- Level Events in the AWT, Low-Level Event Types User Interface Components: Text Input, Choice Components, Menus, Dialog Boxes Setting the Look and Feel of UI, Introduction to JApplet Multithreading: Processes and Threads, Runnable Interface and Thread Class , Thread Objects, Defining and Starting a Thread, Pausing Execution with Sleep, Interrupts, Thread States, Thread Properties, Joins, Synchronization	**6**
5	**Collection and Database Programming** Collections: Collection Interfaces, Concrete Collections- List, Queue, Set, Map, the Collections Framework Database Programming: The Design of JDBC, The Structured Query Language, JDBC Installation, Basic JDBC Programming Concepts, Query Execution, Scrollable and Updatable Result Sets, Metadata, Row Sets, Transactions	**7**
6	**Networking and Web:** Networking: Overview of Networking, Networking Basics, Working with URLs, Creating a URL, Parsing a URL, Reading Directly from a URL, Connecting to a URL, Reading from and Writing to a URL Connection, Sockets, Reading from and Writing to a Socket, Writing the Server Side of a Socket, Datagrams, Writing a Datagram Client and Server. Servlet and JSP: Introduction to Servlet, The servlet Lifecycle, Retrieving Information and Sending Information, Database Connectivity using servlet, Introduction to JSP, Writing Scriplets, The jsp Lifecycle, Retrieving Information and Sending Information, Database Connectivity using jsp	**7**

Term Work:

1. 25 marks for performance in practical and experiments
2. 25 marks for Two Objective Tests each of 25 mark

Practical Examination:

- Minimum 15 experiments should be conducted based on above topics and covering following list. At least two experiments should be conducted on each unit in the syllabus.

1. Create a class called Employee that includes three pieces of information as instance variables- first name, a last name and a monthly salary. Your class should have a constructor that initializes the three instance variables. Provide a set and a get method for each instance variable. If the monthly salary is not positive, set it to 0.0. Write a test application named EmployeeTest that demonstrates class Employee's capabilities. Create two Employee objects and display each object's yearly salary. Then give each Employee a 10% raise and display each Employee's yearly salary again.

2. Create class SavingsAccount. Use a static variable annualInterestRate to store the annual interest rate for all account holders. Each object of the class contains a private instance variable savingsBalance indicating the amount the saver currently has on deposit. Provide method calculateMonthlyInterest to calculate the monthly interest by multiplying the savingsBalance by annualInterestRate divided by 12this interest should be added to savingsBalance. Provide a static method modifyInterestRate that sets the annualInterestRate to a new value

3. Write a program to test class SavingsAccount. Instantiate two savingsAccount objects, saver1 and saver2, with balances of Rs 2000.00 and Rs 3000.00, respectively. Set annualInterestRate to 4%, then calculate the monthly interest and print the new balances for both savers. Then set the annualInterestRate to 5%, calculate the next month's interest and print the new balances for both savers.

4. Create Vehicle Interface with name, maxPassanger, and maxSpeed variables. Create LandVehicle and SeaVehicle Inteface from Vehicle interface. LandVehicle has numWheels variable and drive method. SeaVehicle has displacement variable and launch method. Create Car class from LandVehicle, HoverCraft from LandVehicle and SeaVehicle interface. Also create Ship from SeaVehicle. Provide additional methods in HoverCraft as enterLand and enterSea. Similarly provide other methods for class Car and Ship. Demonstrate all classes in a application.

5. Create abstract class Shape which has instance variables side, area and perimeter

6. And methods calculateArea(), calculatePerimeter() as abstract methods and display() as concrete method. Write subclasses which extend Shape class like Triangle, Rectangle, Circle , Cube and Squere and override abstract methods and display methods in subclass take instance variable if needed as per the formula. And use parameterized constructor to initialize instance variables using "this" reference variable

7. Write Test class and Create a reference variable of Shape which will hold the objects of all the sub classes and calculate respective area, perimeter and display the results.

8. Create the interface stack which has variable size, abstract methods push(),pop(),display(), overflow() and underflow(). We need to implement 3 subclasses IntegerStack, StringStack and DoubleStack respectively by implementing interface. All the methods in interface are declared for string. And in subclass for integerStack convert string to integer. Same thing to all other. Create one test class and check for the working of all the classes.

9. Develop a mathematical package for Statistical operations like Mean, Median, Average, Standard deviation. Create a sub package in the math package -convert. In "convert" package provide classes to

convert decimal to octal, binary, hex and vice-versa. Develop application program to use this package, and build executable jar file of it.

10. Develop application which can handle any 5 combination of predefined compile time and runtime exceptions using multiple catch blocks. Use throws and finally keywords as well.

11. Develop a BankAccount class which should contain all methods of Bank i.e balanceEnquery(), withdraw(), transfer() and deposit(). You should create at least two objects of BankAccount using array and do all operations mentioned above. Also generate user defined exception LowBalanceException, NegetiveNumberException and PasswordMismatchException whenever required. To transfer amount from one account to another use two BankAccount objects.

12. Take file name as input to your program through command line, If file is existing the open and display contents of the file. After displaying contents of file ask user – 1.do you want to add the data at the end of file or 2.replace specified text in file by other text. Based on user's response, then accept data from user and append it to file. If file in not existing then create a fresh new file and store user data into it. Also. User should type exit on new line to stop the program. Do this program using Character stream classes.

13. Take Student information such as name, age, weight, height, city, phone from user and store it in the file using DataOutputStream and FileOutputStream and Retrive data using DataInputStream and FileInputStream and display the result. Use Serialization concept and Bytestream classes.

14. Develop a Swing GUI based standard calculator program. Use event handling, Layout of swing package.

15. Create Stop Watch with Swing GUI and Multithreading. Provide Facility for Lap Counting.

16. Write a program to read a text file one line at a time. Read each line as a String and place that String object into a LinkedList. Print all of the lines in the LinkedList in reverse order.

17. Fill a HashMap with key-value pairs. Print the results to show ordering by hash code. Extract the pairs, sort by key, and place the result into a LinkedHashMap. Show that the insertion order is maintained.

18. Write a GUI based program to create a student registration and Login. Store Registration data in Database and take Login information from Database.

19. Create GUI Based chat application using TCP or UDP.

20. Design a student registration form which contains all the registration details and after registration display success page which should display all the entered details. And also design login form , home page and logout form using servlet.

21. Write a program to store the above form information in database. And while login check with database. Display all student names in home page. Give one search field and button to search for particular student record and display all his information as search result using servlet and jsp using MVC.

Text Books:

1. Core Java- Volume I Fundamentals Cay Horstmann and Gary Cornell, Pearson, Eight edition   Unit 1 to Unit 4
2. Core Java- Volume II Advanced Features Cay Horstmann and Gary Cornell Pearson, Eight edition Unit 5 and Unit 6
3. Java Servlet Programming   Jason Hunter   O'Reilly Publication, 2nd Edition Unit 6
4. Core-Servlet and JavaServer Pages Volume – 1 Marty Hall, Larry Brown, Pearson Education Unit 6

Reference Book:
4. JAVA-The Complete Reference Herbert Schildt Mcgraw Hill, Oracle Press Ninth Edition
5. Head First Java Eric Freeman Elisabeth Robson Bert Bates Kathy Sierra   O'Reilly Publication   3rd edition

6. Head First Servlets and JSP Bryan Basham, Kathy Sierra, Bert Bates O'Reilly Publication 2nd Edition

HM- CS507 –Business English

TEACHING SCHEME	EXAMINATION SCHEME
Theory : --	Term work: 25 Marks
Tutorial :01 Hrs / Week	Theory :
Practical: --	Practical : 25 Marks

Prerequisite: Softskills , English Grammar
Course Objectives:
1. Develop basic skills to deal with people in business situations
2. Increase their knowledge of key business concepts worldwide
3. Write and read basic business reports, faxes, and memos
4. Expand vocabulary related to general business situations
5. Develop confidence to deal with people and basic issues in the business world

Course Outcomes:
Upon successful completion of this course, the student will be able to –
1. Business English courses provide language instruction and training in business communication,
2. This course develops communication skills in a business context and is intended for high intermediate level students interested in using English in business.
3. Students develop language skills for business management and marketing through role-playing, group, discussions.
4. Learner outcomes include improved listening, speaking, and pronunciation skills, as well as business vocabulary development

Unit No.	Unit Name and Contents	No. of Lectures
1	**Getting acquainted with professional culture** i) First day at work ii) Induction program iii) Company Hierarchy **iv)** Behavior Pruning	--
2	**Vocabulary Building and reading** i) Reading or Compression skills ii) Synonyms and antonyms iii) One-word substitution iv) Prefixes or Suffixes v) Idioms and phrases vi) Homonyms or homographs vii) Irregular verbs like (write, wrote, written) viii) Situational of Vocabulary	--

3	**Effective Vocal Communication** i) Effective Meetings ii) Video Conferencing iii) Effective Telephonic Communication iv) Breaking Bad news	--
4	**Effective Written Communication** i) E-mail Writing ii) Business Letters iii) Report Writing iv) Minutes of Meeting v) Resume Writing vi) Memo Writing	--
5	**Public and Presentation Skills** i) Preparing and Conducting Presentation ii) Body Language iii) Overcoming Stage Fear iv) Best Practices' v) Interviewing and Being interviewed	--
6	**Miscellaneous** i) Group Discussion ii) Handling Complains iii) Negotiation Skills iv) Business Etiquettes	--

Term Work:
1. Individual Performance or Presentation to be Evaluated Continuously.
2. Group Activity Performance to be Evaluated in the Batch.
3. Assignments or Writeup (Minimum 10).

Practical Instruction:
1. Students May be evaluated based on practical performance as per list of experiments mentioned in the syllabus.
2. Oral performance may be considered to evaluate and improve their communication skills.
3. Students may be given written tasks to evaluate their business writing skills.

Text Books:
1. Technical Communication by M. Ashraf Rizvi
2. English for Technical Communication by K. R. Laxminarayan, SCITECH 2nd Edition 2014

T. Y. B.Tech (Computer Science and Engineering) Sem – VI

PCC-CS601– Compiler Construction

TEACHING SCHEME	EXAMINATION SCHEME
Theory : 3 Hrs/Week	**Term work:** 25 Marks
Tutorial : --	**Theory** : ESE 70 Marks CIE 30 Marks
Practical: 2 Hrs/Week	**Practical** : --

Prerequisite: C, System Programming, Data structures

Course Objectives:
1. To introduce the fundamentals of compilers and their phases.
2. To design and implement phases of a compiler.
3. To expose the students to various tools like LEX and YACC.

Course Outcomes:
Upon successful completion of this course, the student will be able to –
1. Recall the compiler phases and compiler construction tools like LEX and YACC.
2. To design and implement Lexical Analyser for a simple language.
3. To design and implement Syntax analyzer for a simple expression.
4. To apply Syntax directed translations and Syntax Directed definitions to generate intermediate code.
5. To identify appropriate code optimizing transformation for the given code.
6. To explain concept of code generation.

Unit No.	Unit Name and Contents	No. of Lectures
1	**Introduction** Compilers, Phases of a compiler, Compiler construction tools, cousins of the compiler.	5
2	**Lexical Analysis:** Role of a Lexical analyzer, input buffering, specification and recognition of tokens, finite automata implications, designing a lexical analyzer generator.	6
3	**Syntax Analysis**: Role of Parser, Writing grammars for context free environments, Top-down parsing, Recursive descent and predictive parsers (LL), Bottom-Up parsing, Operator precedence parsing, LR, SLR and LALR parsers	9
4	**Syntax Directed Translation and Intermediate Code Generation**: Syntax directed definitions, construction of syntax tree, S-attributed definitions,	7

	L-attributed definitions, Intermediate languages, assignment statements, back patching.	
5	**Code Optimization**: Principle sources of optimization, optimization of Basic Blocks, loops in flow graphs, Peephole optimization	5
6	**Code Generation**: Issues in design of a code generator and target machine, Run time storage management, Basic blocks and flow graphs, Next use information and simple code generator, Issues of register allocation, code generation from Dags.	7

Term Work:
Minimum of 10 to 12 experiments should be carried out based on the following experiments.

1. Design of preprocessor for C program.

2. Design a complete lexical analyzer for C language.

3. Program to create a symbol table generator.

4. Design a syntax analyzer for simple expression in c language using top down parsing.

5. Program to create a syntax tree for simple expression in c language using recursive descent parsing.

6. To implement intermediate code generator for Boolean expression in three address code format.

7. Implement intermediate code generator for the conditional statements in three address code format.

8. Implement any one bottom up parsing [LR, SLR, LALR, Operator precedence] technique.

9. To implement a program for code generator from labeled tree.

10. Demonstration of compiler and interpreter using LEX and YACC.

Text Books:

1. Compilers - Principles, Techniques and Tools, A.V. Aho, R. Shethi and J.D. Ullman, Pearson Education, ALL Units

Reference Books

1. Compiler construction, D.M. Dhamdare, Mc-Millan

2. LEX & YACC, Dong Brown, John Levine, Tony Mason, O'Reilly, 2nd Edition, Refer for Practical's

PCC- CS602 Operating System

TEACHING SCHEME	EXAMINATION SCHEME
Theory : 04 Hrs / Week	**Term work:** 25 Marks
Tutorial : --	**Theory** : ESE 70 Marks CIE 30 Marks
Practical: 02 Hrs/ Week	**Practical** : --

Prerequisite: Operating System – I, Data Structures and Algorithms

Course Objectives:
1. To provide knowledge to the students about Fundamental architecture of UNIX operating system kernel and algorithms of buffer cache management.
2. To describe Internal File system organizations and related algorithms in UNIX
3. To provide knowledge to the students about System calls for UNIX file system, Process structure, creation and management in UNIX.
4. To provide knowledge to the students about Architecture and algorithms of process scheduling and I/O subsystem as well as memory management.

Course Outcomes:
Upon successful completion of this course, the student will be able to –
1. Describe Fundamental architecture of UNIX operating system kernel and algorithms of buffer cache management.
2. Express the concept of Internal File system organizations and related algorithms.
3. Describe System calls for UNIX file system, Process structure, creation and management in UNIX.
4. Explain Architecture and algorithms of process scheduling and I/O subsystem as well as memory management.

Unit No.	Unit Name and Contents	No. of Lectures
1	**Introduction and buffer cache:-** General Overview of the System - History, System Structure, User Perspective, Operating System Services, Assumption About Hardware, Architecture of UNIX OS, Introduction to system concepts, Kernel Data Structure, System Administration,	7
2	**Buffer Cache: -** Buffer headers, structure of the buffer pool, scenarios for retrieval of a buffer, reading and writing disk blocks, advantages and disadvantages of cache.	6
3	**Internal Representation of Files:-**I-nodes, structure of the regular file, directories, conversion of a pathname to i-node, super block, I-node assignment to a new file, allocation of disk blocks, other file types.	6
4	**System Calls for file system:-** System Calls for file system:- Open, Read, write, File and Record Locking, Adjusting the position of FILE I/O-LSEEK, Close, File Creation, Creation of Special File, Change Directory and Change Root, Change	6

	Owner and Change Mode, Stat and fstat, Pipes, Dup, Mounting and Un-mounting file systems, Link, Unlink, File System Abstractions, File system maintenance	
5	**The Structure of process:-** Process stages and transitions, layout of system memory, the context of a process, Saving context of a process, manipulation of the process address space	6
6	**Process Control and Scheduling:-**Process Control: - Process creation, signals, process termination, awaiting process termination, invoking other programs, the user id of a process, the shell, System Boot and the Init process.	7

Text Books:

1. The design of Unix Operating System - Maurice J. Bach (PHI)

Refence Books:

1. Linux System Programming - Robert Love, Publisher - SPD, O' REILLY
2. Unix concepts and administration – 3rd Edition – Sumitabha Das (TMGH).

PCC- CS603–Database Engineering

TEACHING SCHEME	EXAMINATION SCHEME
Theory : 4 Hrs/Week	**Term work:** 25 Marks
Tutorial : --	**Theory** : ESE 70 Marks CIE 30 Marks
Practical: 2 Hrs/Week	**Practical** : 50 Marks

Prerequisite: Set Theory, Operating System, Data Structures

Course Objectives:
1. To understand fundamental concepts and algorithms of Database Systems.

2. To gain familiarity with SQL and DBMS.

3. To learn database design techniques.

Course Outcomes:
Upon successful completion of this course, the student will be able to –

1. Understand fundamentals of database management systems.

2. Represent logical design of database using E-R Diagram.

3. Analyze & construct good database design.

4. Apply SQL queries to design & manage the database.

5. Understand transactions, concurrency and apply to database system.

6. Understand failures in database and appropriate recovery techniques.

Unit No.	Unit Name and Contents	No. of Lectures
1	**INTRODUCTION TO DATABASES [Text Book 1]:** Database System Applications, Purpose of Database Systems, View of Data, Database Languages, Specialty Databases, Database Users & Administrators, Structure of Relational Databases, Database Schema, Keys, Schema Diagrams, Relational Query Languages, Relational Operations.	8
2	**E-R MODEL AND DATABASE DESIGN [E-R Model: Text Book 1] [Normalization: Text Book 2]** E-R Model: The Entity-Relationship Model, Constraints, Entity-Relationship Diagrams, Reduction to Relational Schemas Normalization: Data Redundancies & Update Anomalies, Functional Dependencies, The Process of Normalization, First Normal Form, Second Normal Form, Third Normal Form, Boyce-Codd Normal Form, Fourth Normal Form, Fifth Normal Form.	10
3	**STRUCTURED QUERY LANGUAGE (SQL) [Text Book 1]:** Overview of the SQL Query Language,SQL Data Definition, Basic Structure of SQL Queries,	8

	Additional Basic Operations, Set Operations, Aggregate Functions, Nested sub Queries, Modification of Databases.	
4	**DATA STORAGE & INDEXING [Text Book 1]:** File Organization,Organization of records in File, Data Dictionary Storage, Database Buffer, Basic Concepts indexing & hashing, Ordered Indices, B+ Tree Index files, Multiple-Key Access, Static Hashing.	10
5	**TRANSACTION MANAGEMENT [Text Book 1]:** Transaction Concept, A Simple Transaction Model, Transaction Atomicity and Durability, Transaction Isolation, Serializability, Lock-Based Protocols, Deadlock Handling, Timestamp-Based Protocols, Validation-Based Protocols.	11
6	**RECOVERY SYSTEM [Text Book 1]:** Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithm, Failure with Loss of Nonvolatile Storage, Remote Backup Systems.	5

Practical :

Minimum 12 -14 Assignments based on the following topics.

1. Draw an E-R Diagram of any organization.

2. Reduce above mentioned E-R Diagram into tables.

3. Normalize any database from first normal form to Boyce-Codd Normal Form (BCNF).

4. Write a program of Database connectivity with any object oriented language.

5. Use DDL Queries to create, alter (add, modify, rename, drop) & drop Tables.

6. Use DML Queries to insert, delete, update & display records of the tables.

7. Create table with integrity constraints like primary key, check, not null and unique.

8. Create table with referential integrity constraints with foreign key, on delete cascade and on delete set null.

9. Display the results of set operations like union, intersections & set difference.

10. Display the results of Join Operations like cross join, self join, inner join, natural join, left outer join, right outer join and full outer join.

11. Display the records using Aggregate functions like min, max, avg, sum & count. Also use group by, having clauses.

12. Display the results using String operations.

13. Create & Update views for any created table.

14. Write java program to implement dense and sparse indexing

15. Write java program to implement B+ tree indexing.

16. Write java program to implement static hashing.

Text Books:

1. Database System Concepts, A. Silberschatz, H.F. Korth, S. Sudarshan, 6th Edition, Mc Graw Hill Education.
2. Database Systems - A practical approach to Design, Implementation and Management Thomos Connolly, Carolyn Begg, 3rd Edition, Pearson Education

Reference Books:

1. Database Systems – Design, Implementation and Management,Rob & Coronel   5th Edition,Thomson Course Technology
2. Fundamentals of Database Systems, Ramez Elmasri, Shamkant B. Navathe, 4th Edition,Pearson Education

PCC- CS604 Machine Learning

TEACHING SCHEME	EXAMINATION SCHEME
Theory : 03 Hrs. / Week	**Term work: --**
Tutorial :01 Hrs. / Week	**Theory : ESE 70 Marks** CIE 30 Marks
Practical: --	**Practical : --**

Prerequisite: Discrete Mathematics, Mathematics

Course Objectives:
1. To understand Human learning aspects.
2. To understand primitives in learning process by computer.
3. To understand nature of problems solved with Machine Learning

Course Outcomes:
Upon successful completion of this course, the student will be able to –
1. Explain machine learning concepts.
2. Analyze the Machine learning model.
3. Design solution using machine learning techniques.

Unit No.	Unit Name and Contents	No. of Lectures
1	**Introduction to Machine Learning:** Definition, Terminology, Types of learning, Machine Learning Problem categories, Machine learning architecture, process, Lifecyle, Performance measures,tools and framework, data visualization.	6
2	**Regression:** Simple regression – hypothesis, cost function, parameter learning with gradient descent, learning rate,Gradient Descent for linear regression, examples, simple regression in matrix form. Multivariate Linear Regression – multiple features, hypothesis functions, Gradient Descent for multiple variables, Feature scaling, polynomial regression	8
3	**Classification- logistic regression & Neural Network:** Definition, logistic regression – hypothesis representation, decision boundary, cost function, gradient descent for logistic regression. multiclass classification, Regularization - Overfitting & Underfitting, cost function, Regularized Linear Regression, Regularized Logistic Regression Neural Networks- Neuron representation and model, Hypothesis for neuron, cost function, solution of a problem using single neuron. Gradient descent for a neuron. Neural network, Multiclass classification with neural network. Learning in neural network-backpropagation algorithm	8
4	**Classification- Decision trees and Naïve Bayes**	8

	Decision trees: definition, terminology, the need, advantages, and limitations. constructing and understanding Decision trees, common problems with Decision trees, Decision tree algorithms, random forest, examples. Conditional probability and Naïve Bayes Classifier Instance-based classifier – K- Nearest Neighbour Classifier	
5	**Unsupervised learning:** Clustering, K Means clustering, Hierarchical clustering, Association Rule mining	6
6	**Recommendation System and Time series analysis** Basic Text Processing with Python, regular expression, Natural Language Processing, Text Classification, Topic modeling Popularity based recommender engines, Content based recommendation engines, Classification based recommendation engine, collaborative filtering Date and Time Handling, Window functions, Correlation, Time Series Forecasting	6

Text Books:

1. Machine Learning with Python- an approach to applied ML, by Abhishek Vijayvargia, BPB publications
2. Practical Machine Learning by Sunila Gollapudi Packt Publishing Ltd.
3. Machine Learning by Tom M. Mitchell, McGraw Hill Education; First edition

Reference Book:

1. Machine Learning for dummies John Paul Muller, Willey Publication
2. EthemAlpaydin : Introduction to Machine Learning, PHI 2nd Edition-2013

OEC- CS605Data Structure using C

TEACHING SCHEME	EXAMINATION SCHEME
Theory : 03 Hrs. / Week	Term work: --
Tutorial :--	Theory : ESE 70 Marks CIE 30 Marks
Practical: --	Practical : --

Prerequisite: Discrete Mathematics, Mathematics

Course Objectives:
1. To learn the features of C
2. To learn the linear and non-linear data structures
3. To explore the applications of linear and non-linear data structures
4. To learn to represent data using graph data structure
5. To learn the basic sorting and searching algorithms

Course Outcomes:
Upon successful completion of this course, the student will be able to –
1. Explain basics of c programming
2. Describe linear and non-linear data sctrues
3. Describe data using graph
4. Explain different sort and searching methods

Unit No.	Unit Name and Contents	No. of Lectures
1	Structure of a C program – compilation and linking processes – Constants, Variables – Data Types – Expressions using operators in C – Managing Input and Output operations – Decision Making and Branching – Looping statements	6
2	. Arrays – Initialization – Declaration – One dimensional and Two-dimensional arrays. Strings- String operations – String Arrays. Simple programs- sorting-searching – matrix operations.	8
3	Functions – Pass by value – Pass by reference – Recursion – Pointers – Definition – Initialization – Pointers arithmetic. Structures and unions – definition – Structure within a structure – Union – Programs using structures and Unions – Storage classes, Pre-processor directives.	8
4	Arrays and its representations – Stacks and Queues – Linked lists – Linked list-based implementation of Stacks and Queues – Evaluation of Expressions – Linked list based polynomial addition.	8
5	Trees – Binary Trees – Binary tree representation and traversals –Binary Search Trees – Applications of trees. Set representations – Union-Find operations. Graph and its representations – Graph Traversals.	6
6	Linear Search – Binary Search. Bubble Sort, Insertion sort – Merge sort – Quick sort – Hash tables – Overflow handling.	6

Text Books:

1. Pradip Dey and Manas Ghosh, ―Programming in C, Second Edition, Oxford University Press.
2. Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, ―Fundamentals of Data Structures in C, Second Edition, University Press.

Reference Book:

1. Robert Kruse, C.L.Tondo, Bruce Leung, Shashi Mogalla , ― Data Structures and Program Design in C, Second Edition, Pearson Education.
2. Jean-Paul Tremblay and Paul G. Sorenson, ―An Introduction to Data Structures with Applications, Second Edition, Tata McGraw-Hill.

PCC- CS606 C# Programming

TEACHING SCHEME	EXAMINATION SCHEME
Theory : 02 Hrs./Week	**Term work:** 25 Marks
Tutorial : --	**Theory** :--
Practical: 02 Hrs./Week	**Practical** : 50 Marks

Prerequisite: C++, Java, HTML

Course Objectives:

1. Understand code solutions and compile C# projects within the .NET framework

2. Demonstrate knowledge of object-oriented concepts Design user experience and functional requirements C#.NET application

3. Create and manipulate GUI components in C# and Design and Implement database connectivity using ADO.NET in window based application

4. Create network based and multithreaded applications using C#

5. Creating ASP.Net applications using standard .net controls

6. Maintain session and controls related information for user used in multi-user web applications

Course Outcomes:

Upon successful completion of this course, the student will be able to –

1. Develop correct, well-documented programs using the C# programming language.

2. Learn to develop object-oriented programs using C# classes and objects

3. Learn to use Windows Forms and WPF to create GUI-based programs

4. Build networking and multithreading based programs using C#

5. Design web applications using ASP.NET use ASP.NET controls in web applications.

6. Debug and deploy ASP.NET web applications and create database driven ASP.NET web applications and web services

Unit No.	Unit Name and Contents	No. of Lectures
1	.Net architecture and C# Basics: .Net Architecture: The Relationship of C# to .NET, The Common Language Runtime, A Closer Look atIntermediate Language, Assemblies, .NET Framework Classes, Namespaces, C# basics: Predefined data types, TheMain () Method,More on Compiling C#	3

	Files, Console I/O, Using Comments# ProgrammingGuidelines, Dynamic variables, DLL creation & calling.	
2	Object oriented programming in C# : Classes and Structs, Class Members, Anonymous Types, Structs, Partial Classes, StaticClasses, The Object Class, Extension Methods, Inheritance: Types of Inheritance, Implementation Inheritance, Modifiers, Interfaces	4
3	Arrays, Operators and casts and strings: Arrays: Simple Arrays, Multidimensional Arrays, Jagged Arrays, Array Class, Array andCollection Interfaces, Enumerations Operators and casts: Operators, Type Safety, Comparing Objects for Equality, Operator Overloading, User-Defined Casts, String:System. String, Building Strings, String Builder Members, Format Strings, Regular Expressions.	4
4	Windows Form & Database with ADO.NET: Introduction to GUI application & components –add data control programmatically, Link data tocontrol, process all control, track the visible forms, Find all MDI child forms, Save configurationsetting for form, Force list box to scroll items, Restrict text box, Use of auto complete combobox ,Sort a list view, Database with ADO.NET-Overview of Ado.NET, Data components inVisual Studio .NET.	5
5	Threading and Networking: Threading: Overview, Asynchronous Delegates, the Thread Class and Thread Pools, Threading Issues,Synchronization, Timers. Networking: Networking-Obtain information about Local network, Detect changes in network, Downloaddata over HTTP or FTP, Download a File & Process using Stream, Respond to HTTP requestfrom your application.	4
6	Introduction to ASP.NET 4.5 ONE ASP.NET: Introducing One ASP.NET, Simplifying a Complex Ecosystem, How Do You Benefit? ASP.NET web form structure: Application Location Options, The ASP.NET Page Structure Options, ASP.NET 4.5 Page Directives, ASP.NET Page Events, Dealing with Postbacks, Cross-Page Posting ,ASP.NET Application Folders, Compilation, Build Providers, Global.aspx	4

Term Work:
1. It should consist of 10 to 12 experiments based on the above syllabus covering following list of assignments

Practical Examination (Instructuion):
1. 50% of the experiment should be console based & 40 % experiment should be windows form application. and 10% should be web based application)

Sample Experiments

1. Language Introduction (Includes console based application, creation of dll, running a program without IDE) calling a method from another program.
2. OOPS concepts in C#-Class, Implementation Inheritance, Extension methods (Use Any application).
3. Develop DLL file and use it in application program. (Use Any application)
4. Implementation of Interface Inheritance (Use Any Application).
5. Implementation of Multidimensional & Jagged array (Use Any application).
6. Use of properties in any application.
7. Implementation of Operator overloading (Any application).
8. String manipulation using String & String builder(Any application)
9. Develop program to use Regex.Matches method and Regular Expression pattern matching.
10. Design a Windows Form based application for different controls.(Any application)
11. Design a Windows Form based MDI application with different controls.(Any application)
12. Design a Windows Form based application for field validation.(Any application)
13. Design a any Windows Form based application with Database connectivity with all field validation.(Any application)
14. Develop a Windows Form application that performs SELECT, INSERT, UPDAE & DELETE queries and also displays the List of Books available in a Library System by fetching the details from a database. The C# application must also contain the filter capability.
15. Implement console based networking application to obtain information of network & detect changes in network.
16. Design a Windows form application to download file & process it using stream.
17. Simple ASP.NET web application deployment in IIS server
18. Design simple login and registration page using client side validation controls in ASP.NET
19. Do the server side validation by using database connectivity for above problem and display all the records of database when successfully logged in using ASP.NET.

Text Books:
1. Professional C# 2012 & .Net 4.5 by Christian Nagel, Bill, Evjen, Jay Glynn, Morgan Skinner, Karli Watson Wrox Publication, 1 to 2 and unit 3 (threading)
2. A Programmer's Guide to ADO.Net in C# by Mahaesh Chand, Apress Publication Unit 3 Windows app and ADO.NET

3. Visual C# 2010 Recipes- A Problem-Solution Approach by Allen Jones , Adam Freeman , Matthew MacDonald , RakeshRajan Apress Publication Unit 3 Windows app and ADO.Net and Unit 4 networking
4. Professional ASP.NET 4.5 in C# and VB   John Wiley & SonsInc. WROX publication Unit 4 ASP part and Unit 5 and 6

Reference Books:

1. ASP.NET 4.5- Black book by Kogent Dreamtech Publication

PW- CS607Domain Specific Mini-project

TEACHING SCHEME	EXAMINATION SCHEME
Theory : --	**Term work:** 25 Marks
Tutorial : --	**Theory** :--
Practical:02 Hrs/Week	**Practical** : 50 Marks

Prerequisite: --C ,C++, Java programming

Course Objectives:
1. To expose the students to use engineering approach to solve domain specific real time problem.
2. To use the appropriate and newer technologies while developing the project.
3. To learn the skills of team building and team work.

Course Outcomes:
Upon successful completion of this course, the student will be able to –
1. Identify specific problem statement from a selected domain.
2. Analyse the problem and prepare SRS and design document.
3. Write code and carry out testing.
4. Write a report covering details of the project and give presentation on a project.

Contents

The students should form group of maximum 5 students and every group is supposed to choose a specific domain to do the mini project-II. Further the group should identify the relevant problem in the selected domain and propose the solution, which can be implemented as a mini-project using suitable technology. The mini-project-II work should be evaluated by a team of teachers appointed by the department/COE. The evaluation and marking should include Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) during which the group should give presentation and demonstration of their work done. Care should be taken to avoid out-sourcing of the work.