Shivaji University, Kolhapur.

Revised Syllabus And structure of
T.E. Part-I & II
(Information Technology Engineering)
Semester III and IV
(w.e.f. Academic Year 2015-16)

SHIVAJI UNIVERSITY, KOLHAPUR

(To be implemented from Academic Year 2015-16)

Class : TE Part I (Semester V) **Branch : Information Technology**

Name of Subject	Teaching Scheme Per				Examination Scheme					
	L	T	P	Total	Theory Paper		T/W	OE	POE	Total
					Written	Online				
1. Operating System-I *	3	1	-	4	50	50	25	-	-	125
2. Database Engineering *	4	-	2	6	50	50	25	-	50	175
3. Computer Algorithms	4	-	-	4	100	-	-	-	-	100
4. System Programming	3	-	-	3	100	-	-	-	-	100
5.Object Oriented Modeling and Design	3	1	-	4	100	-	25	-	-	125
6. Application Development Tool I	2	-	4	6	-	-	50	-	50	100
7. Soft skill	-	-	2	2	-	-	25	50	-	75
Total	19	2	8	29	400	100	150	50	100	800

Class : TE Part II (Semester VI)

Name of Subject	Teaching Scheme Per				Examination Scheme					
	L	T	P	Total	Theory Paper		T/W	OE	POE	Total
					Written	Online				
1. Computer Graphics*	3	-	2	5	50	50	25	-	-	125
2. Information Security	3	1	-	4	100	-	25	-	-	125
3. Internet Technology*	3	-	2	5	50	50	25	-	50	175
4. Operating System II	3	-	2	5	100	-	25	-	-	125
5. Software Testing & Quality Assurance	3	-	-	3	100	-	-	-	-	100
6. Application Development Tool II	2	-	4	6	-	-	50	-	50	100
7. Seminar	-	1	-	1	-	-	50	-	-	50
Total	17	2	10	29	400	100	200	-	100	800

*** Note for Online Examinations -:**

1. The examination will be having two part viz. part A & part B.

    i. Part-A: 50 marks theory paper similar to the existing theory paper exam. The nature of the questions will be descriptive, analytical and problem solving.

    ii. Part-B: 50 marks computer based exam with multiple choice questions (MCQs) .

2. The marks obtained in the individual heads should be added and considered as marks of the respective theory paper out of 100 marks.

3. The questions of part-A and part-B will be based on the entire syllabus of the respective subjects.

4.The questions in part-B will be of 1 or 2 marks only.

6. Duration of part-A exam will of 2 hours and that of part-B will be of 1 hour.

7. No separate passing head for part-A and part-B.

8. The scheme of revaluation is not applicable for part-B, however is applicable for part-A

T.E. Information Technology Semester-V (Revised)

1. OPERATING SYSTEM-I

Lecture : 3Hrs/week **Tutorial : 1 Hr/week** **Theory : 50 marks**
Online : 50 marks **Term Work : 25 marks**

Prerequisites

1) Basic knowledge of digital system and microprocessors, memory, interrupts is essential.

Course Objectives:

1) To introduce Operating systems, types and its use
2) To introduce process, threads and their management,
3) To introduce process and Thread scheduling, interprocess synchronization and communication
4) To introduce memory management
5) To introduce input output devices & their management

Unit 1: Introduction to Operating Systems (8)

Introduction to Operating Systems, System structures: What operating systems do; Computer System organization; Computer System architecture; Operating System structure; Operating System operations; Types of Operating Systems*, Distributed system; Special-purpose systems; Operating System Services; User - Operating System interface; System calls; Types of system calls; System programs; Operating System structure; Virtual machines; System boot.

Unit 2: Process Management (8)

Process concept; Process scheduling; Operations on processes; Inter-process communication. Multi-Threaded Programming: Overview; Multithreading models; Thread Libraries; Threading issues. Process Scheduling: Basic concepts; Scheduling criteria; Scheduling algorithms; Multiple-Processor scheduling; Thread scheduling.

Unit 3: Process Synchronization (7)

Synchronization: The Critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors.

Unit 4: Deadlocks (6)

System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock.

Unit 5: Memory Management: **(7)**

Memory Management Strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. Virtual Memory Management: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing

Unit 6: IO Systems **(4)**

Overview, I/O Hardware, Application I/O Interface, Kernel IO Subsystem, Transforming I/O Request to Hardware Operations, Streams

Text Books :

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 8th edition, Wiley India, 2009

Reference Books:

1. Operating Systems –Concepts and design –Milan Milenkovic (TMGH)

(For Types of Operating Systems*-Refer Chapter 1 in Operating Systems –Concepts and design –Milan Milenkovic (TMGH))

2. Operating Systems: Internals and Design Principles (8th Edition)- by William Stallings(Pearson Education International)

3. Modern Operating Systems by Andrew S. Tanenbaum (Pearson Education International)

Term Work:

Tutorial work to be considered for awarding termwork marks.

In tutorial session Students are expected to solve problems based on process scheduling, process synchronization and communication, deadlocks, memory management techniques

Students are also expected to write programs for process management and process synchronization

DATABASE ENGINEERING

Lectures: 4 Hrs/week	**Theory: 50 Marks**	**Online: 50 Marks**
Practical: 2 Hrs/week	**Term Work: 25 Marks**	**POE: 50 Marks**

Prerequisites

1. Basic knowledge of data structures is essential

Course Objectives:

1. To understand the fundamental concepts of database management.
2. To give systematic database design approach.
3. To understand the basics of transaction processing and concurrency control in database systems.

Unit 1. Introduction: (8)

Purpose of Database Systems, View of Data, Data Models, Database Users and Administrators, Overall System Design, **Entity Relationship Model-** Basic Concepts, Constraints, Keys, E-R Diagram, Weak Entity Sets, Specialization, Generalization, Aggregation, Alternative ER Notations, Reducing E-R Diagrams to Tables.

Unit 2. Relational Model: (10)

Structure of Relational Databases, the Relational Algebra, Tuple Relational Calculus, Structured Query Language (SQL), PL/SQL- Stored Procedures, functions, trigger, cursor

Unit 3. Integrity Constraints and Design: (7)

Domain Constraints, Referential Integrity, Functional Dependencies, Closure of set of Functional Dependencies, Canonical cover, Pitfalls in Relational database Design, Decomposition, Desirable Properties of Decomposition, Normalization using Functional Dependencies (1NF, 2NF, BCNF, 3NF).

Unit 4. File and Index Structure: (9)

Physical storage media, Storage access, File Organization, Organization of Records in Files, Data Dictionary Storage, Indexing and Hashing: Basic Concepts, Ordered Indices, B+ Tree Index Files, B-Tree Index Files, Static Hashing, Dynamic Hashing, Comparison of Indexing &Hashing, Multiple Key Access- Grid Files

Unit 5. Concurrency Control and Crash Recovery (9)

Transaction concept, Transaction state, Concurrent Executions, Serializability, Recoverability, Testing for Serializability, Lock-Based Protocols, Graph based Protocols, Time-Stamp Based Protocols, Validation based protocols, Multiple Granularity, Failure Classification, Recovery and Atomicity, Log-Based Recovery, Checkpoints, Shadow Paging, Buffer Management.

Unit 6. Database Security and Authorization: (5)

Access Control, Discretionary Access Control, Mandatory access control, Additional Issues Related to security

Text Book:

1. Database System Concept by Henry F. Korth, Abraham Silberschatz, Sudarshan (McGraw Hill Inc.) **Fourth Edition onwards** (Unit 1 to 5)

2. Database Management System – RamKrishnan, Gehrke (McGraw Hill Inc.)

Third Edition (Unit 6)

Reference Books:

1. Principles of DataBase Systems by J.D. Ullman (Galgotia Publications)

2. Database Design by Wiederhold (McGraw Hill Inc.)

3. Fundamentals of Database Systems- Elmasri, Navathe (BengaminCummings, 1989).

Term Work:

It should consist of minimum 10 experiments based on above topics and should be implemented as per the note given below.

Set of Experiment is listed below:

1. Study and design of ER diagram
2. Reduction of ER diagrams.
3. Study of DDL. (Create table with all constraints, Alter table, Drop table)
4. Study of DML. (Basic SQL structure-select, from, where clause. Other DML clauses like insert, update, delete, in, between, etc.)
5. Database joins- (Natural Join, outer joins.)
6. String, Set operations, Order by clause. Queries based on above commands.
7. Aggregate functions, Group by, Having clauses.
8. Study of DCL commands (Grant, Revoke)
9. Creation and use of Views
10. Introduction to PL/SQL- Stored Procedures, functions, trigger, cursor
11. Study of Functional dependency, Closure & Canonical Cover. Implementation of closure of Functional dependencies and canonical cover.
12. Study of Normalization & Normal forms
13. Study of B+ index file (creation, traversal, deletion operations)
14. Implementation of static index structure.
15. Simulation of Deferred Log based recovery scheme.
16. Simulation of Immediate Log based recovery scheme.
17. Implementation of database connectivity using JDBC-ODBC

Note:

1. Experiments 1 and 2 and 11 to 12 are for demonstration and understanding of database designing and other concepts.
2. Experiments no. 3 to 10 should be implemented using RDBMS Package.
3. Experiments 14 to 17 are to be implemented using programming language and RDBMS (if required) .

COMPUTER ALGORITHMS

Lectures: 4 hrs/week **Theory: 100 marks**

Prerequisites: Data Structures, numerical problem solving methods.

Objectives:

1. To analyze the asymptotic performance of algorithms.
2. To analyze the time and space complexity of simple algorithms and recursively defined algorithms.
3. To study how to design good algorithms.
4. To study major algorithms and data structures
5. To study important algorithmic design paradigms and methods of analysis.

Unit 1: Introduction **(6)**

What is algorithm, Algorithm Specification, Recurrence relations, Performance Analysis, Randomized Algorithms.

Unit 2: Divide and Conquer **(7)**

The general method, Binary search, Finding the maximum and minimum, Merge sort, Quick sort, Selection sort and analysis of these algorithms.

Unit 3: The Greedy method **(7)**

The 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, Single source shortest paths.

Unit 4: Dynamic Programming **(9)**

The general method, Multistage graphs, All pair shortest paths, Optimal binary search trees, 0/1 knapsack, Reliability design, Traveling Sales person problem.

Unit 5: Basic Traversal and Search Techniques **(7)**

Techniques for Binary Trees, Techniques for Graphs – Breadth First Search& Traversal, Depth First Search & Traversal, Connected components and Spanning Trees; Bi-connected components and depth first search.

Unit 6:

Backtracking **(7)**

The general method, 8-queen problem, sum of subsets, Knapsack Problem, Hamiltonian Cycle, Graph Coloring

NP Hard and NP Complete Problems **(3)**

Basic Concepts, Introduction to NP Hard Graph Problems.

Text Books:

1. Fundamentals of Computer Algorithms - Ellis Horowitz, Sartaj Sahani, Sanguthevar Rajasekaran, Universities Press, Second Edition.

References:

1. Introduction to Algorithms - Thomas Cormen, Charles Leiserson, Ronald Rivest, Clifford Stein, PHI, Third Edition
2. Essential Algorithms: A Practical Approach to Computer Algorithms, Rod Stephens, Wiley International .

2. SYSTEM PROGRAMMING

Lecture: 3 Hrs/Week **Theory: 100 Marks**

Prerequisites:

1) Basic knowledge of data structures, assembly language programming ,microprocessors.

Course Objectives:

1) To provide knowledge of major system software used in a computer systems & language processing activities involved
2) To provide knowledge of language processors and various language processing activities
3) To compare and differentiate between language processors.
4) To understand fundamentals of assemblers and macro preprocessors.
5) To understand linkers, loaders, compilers and interpreters
6) To analyze and use different open sources software

Unit 1: Introduction and Overview of Language Processors (6)
What is system software, Goals, System programs and System programming, Wonderland of system software, Views of System Software, Programming languages and Language Processors, Language processing activities, Fundamentals of language processing, Symbol tables

Unit 2: Assemblers (7)
Elements of assembly language programming, a simple assembly scheme, pass structure of assemblers, design of a two pass assembler, a single pass assembler for X86 Family Processors

Unit 3: Macros and Macro Preprocessor (7)
Introduction, Macro definition and call, Macro Expansion, Nested macro calls, Advanced macro facilities, Design of macro preprocessor

Unit 4: Linkers and Loaders (7)
Introduction, Relocation and linking concepts, Design of a linker, Self-relocating programs, Linking in MS DOS, Linking of overlay structured programs, Dynamic linking, Loaders

Unit 5: Compilers and Interpreters (11)
Causes of large semantic gap, Binding and binding times, Data structures used in compilers, Scope rules, Memory allocation, Compilation of expressions, Compilation of control structures, Code optimization, Interpreters: Benefits of interpretation, Overview of interpretation, the java language environment

Unit 6: Open Source Software (2)
LEX, YACC, GCC, GDB, DDD

Text Books:
1.Systems Programming: D.M. Dhamdhere, McGraw Hill, 1st Edition

Reference Books:
1.Systems Programming & Operating systems: D.M Dhamdhere, 2nd Edition (TMH)
2.Systems Programming: J.J. Donavan – (TMH)
3.System Programming , Srimanta Pal , Oxford University Press.

Open Source Resources
gcc.gnu.org
www.gnu.org/s/gdb/
www.gnu.org/software/ddd/

OBJECT ORIENTED MODELING AND DESIGN

Lecture: 3 Hrs/Week **Theory: 100 marks**
Tutorial: 1 Hr/Week **Term Work: 25 marks**

Prerequisites
1) Knowledge of Software Engineering concepts and object oriented concepts

Course Objectives:
1) To provide an understanding of fundamental concept of object modeling.
2) To provide an understanding about class modeling and state modeling.
3) To provide an understanding of Interaction modeling
4) To provide Knowledge about design methodology

Unit 1: Introduction, Modeling Concepts, class Modeling: (7)
What is Object Orientation? What is OO development? OO themes; Modeling as Design Technique: Modeling; abstraction; The three models. Class Modeling: Object and class concepts; Link and associations concepts; Generalization and inheritance; A sample class model; Navigation of class models;

Unit 2: Advanced Class Modeling, State Modeling (6)
Advanced Class Modeling: Advanced object and class concepts; Association ends; N-ary associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived data; Packages; State Modeling: Events, States, Transitions and Conditions; State diagrams; State diagram behavior;

Unit 3: Advanced State Modeling, Interaction Modeling (6)
Advanced State Modeling: Nested state diagrams; Nested states; Signal generalization; Concurrency; A sample state model; Relation of class and state models; Interaction Modeling: Use case models; Sequence models; Activity models. Use case relationships; Procedural sequence models; Special constructs for activity models.

Unit 4: Process Overview, System Conception, Domain Analysis (8)
Process Overview: Development stages; Development life cycle. System Conception: Devising a system concept; Elaborating a concept; Preparing a problem statement. Domain Analysis: Overview of analysis; Domain class model; Domain state model; Domain interaction model; Iterating the analysis.

Unit 5: Application Analysis, System Design: (7)
Application Analysis: Application interaction model; Application class model; Application state model; Adding operations. System design :Overview of system design; Estimating performance; Making a reuse plan; Breaking a system in to sub-systems; Identifying concurrency; Allocation of sub-systems; Management of data storage; Handling global resources; Choosing a soft ware control strategy; Handling boundary conditions; Setting the trade-off priorities; Architecture of the ATM system as the example.

Unit 6: Class Design, Implementation Modeling (6)
Class Design: Overview of class design; Bridging the gap; Realizing use cases; Designing algorithms; Recursing downward, Refactoring; Design optimization; Reification of behavior; Adjustment of inheritance; Organizing a class design; ATM example. Implementation Modeling: Overview of implementation; Fine-tuning classes; Fine-tuning generalizations; Realizing associations; Testing.

Text Books:

1. Object Oriented Modeling and Design with UML2.0 – James Rumbaugh, Michael Blaha, (Pearson second Edition).
2. Software Modeling & Design , Hassan Gomaa, Cambridge University Press.

Reference Books:

1. Object Oriented Analysis & Design- Atul Kahate (McGraw-Hill)
2. Object-Oriented Analysis & design understanding system development with UML 2.0, Mike O'Docherty
3. Object-Oriented Analysis and Design with Applications- Grady Booch, Pearson Education,

Guidelines for conducting tutorials:

- Tutorials must be analysis and design oriented
- Students in a batch shall be divided in a group of 3-4 students for the purpose of tutorial.
- Case studies shall be based on the text books and reference book given in the syllabus.
- Use open source tools for Design like Dia.

The performance of a student in a batch shall be periodically assessed by the concerned batch teacher. The assessment will be considered for determining term work marks.

3. APPLICATION DEVELOPMENT TOOL – I

Lecture: 2 Hrs/Week **Practical: 4 Hrs/Week**
Term Work: 50 Marks **POE: 50 Marks**

Prerequisites:

1) Knowledge of basic C programming , concepts of object orientation.

Course Objectives:

1) To understand differences between C++ and JAVA
2) To understand fundamental concepts like objects, classes, interfaces and polymorphism and its implementation in JAVA
3) To understand robust GUI applications using event handling and Swing with proper exception handling
4) To understand I/O concepts, database connectivity and collections framework in JAVA
5) To Identify classes, objects, members of a class and the relationships among them needed for a specific problem

Unit 1: **(6)**

Introduction: Overview of Java, Java buzzwords, Difference between C++ & Java, Data Types, Arrays, Command line Arguments

Classes: The Object class, Object Construction, Garbage Collection, Nested & Inner classes, String class, Wrapper classes, Class Design Hints

Inheritance: Member Access, Super keyword, final keyword, Abstract Classes, Access Protection, Interfaces, Design Hints for Inheritance

Packages: Defining a package, Searching packages and setting CLASSPATH

Unit 2: **(7)**

Exceptions: Dealing with Errors, Catching Exceptions, Tips for Using Exceptions

I/O: Streams, Text Input and Output, Reading and Writing Binary data
Multi-Threading: What are threads?, Interrupting threads, Thread states, Thread properties and synchronization

Unit 3: **(6)**

Swing : Introducing AWT and Swing, Creating a Frame, Positioning a Frame, Displaying Information in a Component, Introduction to Layout Management, Text Input , Choice Components, Menus, Dialog Boxes

Event Handling: Basics of Event Handling, Mouse Events, the AWT Event Hierarchy

Unit 4: **(3)**

Generic Programming: Why Generic Programming?, Definition of a Simple Generic Class, Generic Methods

Collections: Collection Interfaces, Concrete Collections, the Collections Framework

Unit 5: **(3)**

Database Programming: Design of JDBC, JDBC Configuration, Executing SQL statements

Unit 6: **(1)**

Deploying Applications and Applets: JAR Files, Applets

Text Books:

1. Core Java- Volume I Fundamentals: Cay Horstmann and Gary Cornell, Pearson, Eight edition
2. Core Java- Volume II Advanced Features: Cay Horstmann and Gary Cornell, Pearson, Eight edition (Unit 2 I/O and Unit 5)

Reference Books:

1. JAVA-The Complete Reference: Herbert Schildt, Oracle Press, Mcgraw Hill, Ninth edition
2. A Programmer's guide to JAVA SCJP Certification: Khaleed Mughal and Rolf W. Rasmussen, Addison Wesley, Third edition
3. An introduction to Programming through C++ Abhiram G. Ranade, McGrawHill
4. www.spokentutorial.org, NMEICT Project of MHRD Govt Of India & IIT Bombay.

Guidelines for term work distribution

The distribution of the term work marks is as follows

- 25 marks for performance in practical and experiments
- 25 marks for mini-project to be developed in Java.

Guidelines for experiment list

The experiment list should consist of minimum 15 practical assignments on the above topics. Each experiment should be a problem statement which can be solved using some features of Java. Sample experiment list is given below

1) Installation of jdk on Linux
2) Write a program to implement vector class (Understanding basic structure of java programs)

3) Write a program to implement matrix class (Understanding basic structure of java programs)
4) Write program to implement given inheritance hierarchy (Understanding of inheritance concept)
5) Write a program to create linked list through interface. (Understanding of Interface)
6) Create a Mymath package that will have following features. a. Trigonometric functions (sine, cosine, tangent, secant, cosecant and cotangent) that accepts input in degrees instead of radians. b. Performs Statistical operations like min, max, count, sum and average(Understanding of package)
7) Write a program to create applet and perform the slideshow of images using Multi-threading(Multi-threading)
8) Write a program to remove whitespaces from a text file. Name of the file is given using command line (Understanding of basic IO concepts, command line arguments and exception handling))
9) Write a program to merge and sort data from different files in a single file. (I/O concepts and exception handling)
10) Write a program to copy text from one text box to another on a button click. (Swing and event handling)
11) Write a program to create a GUI student registration form. (Swing controls and event handling)
12) Write a program to demonstrate key and mouse event handling (Event handling)
13) Write a program to demonstrate various methods of ArrayList class. (Collections)
14) Write a program to store and retrieve, delete and update Student's information in Database.(Implementation of database connectivity in java)
15) Study of frame works like stud, spring hibernate etc.

Guidelines for Mini-project

Three students (Maximum) in a group will carry out a mini project.

A batch of practical should be divided into mini project groups. The faculty should guide the project group for selection of the topic and the work to be done. (Topics preferably data structure algorithm simulation using graphics and thread and other concepts).The mini project should consist of defining the problem, analyzing, designing the solution and implementing it using Java (preferably IDE should be used). The faculty shall monitor the progress periodically. A presentation based on the above work is to be given by the group at the end of the semester. The work will be jointly assessed by a team of faculty from the department. A hard copy of project report, along with a softcopy of the programs is to be submitted to the department.

4. Soft Skills

Practical: 2 hrs/week

Term work : 25 Marks **Oral Exam : 50 Marks**

The objective of this course is to enable students to acquire and enhance communication and professional skills required for personality development, corporate business and entrepreneurship skills. During the tutorial and practical sessions, it is expected that the contents of all modules should be delivered to the students of different batches and assignments be given based on the activities discussed as per the modules. Evaluation of the term work should be done on continuous basis and two tests (mid term and end term tests) should be conducted. Students must demonstrate the acquired skills by means of giving presentations, group discussions, interviews, etc. The modules proposed are as under.

1. Art of communication
Communication Theory
Barriers and Filters
Active Listening
Non Verbal Communication
Feedback and Response
Body Language

2. Hidden data of communication
Dealing with feelings
Assertiveness
Self – confidence
Emotional Intelligence

3. World of teams
Team concept
Elements of team work
Formation of a team
Team based activities

4. Adapting to corporate life
Corporate Grooming and dressing
Business Etiquette
Business Ethics
Dinning Etiquette
Ethics policy

5. Discussions, decisions and presentations
What are group discussions
Types of Group Discussions
Presentations
Decision making
Interview Skills
Resume Writing.

6. Job Interview : The Gateway to the Job Market

Types of Interviews, Importance of body language, Need of proper articulation, Probable interview questions, Telephonic or Video Interview.

Text Books :

1. Soft -skills Manual, Infosys Campus connect Program

2. Personality Development and Soft- Skills , Barun K. Mitra ,Oxford University Press.

T.E. Information Technology Part-II (Semester-VI) (Revised)

1. COMPUTER GRAPHICS

Lectures: 3 hrs / week **Theory: 50 marks**
Practical: 2 hrs / week **Online Exam: 50 marks**
Term work: 25 marks

Prerequisites: Basic knowledge of C language and data structures, geometric constructions.

Course Objectives:

1. To understand basics of computer graphics & graphics devices
2. To give more emphasis on implementation aspect of Geometric Transformations and Computer Graphics Algorithm.
3. To prepare the student for advance courses like multimedia / Computer Vision.

Unit 1: Basic Concepts and Graphics Devices: (6)

Introduction to computer graphics, Applications of computer graphics, Pixel, Frame Buffer, Resolution, aspect ratio. **Video display devices:** CRT (Raster-Scan and Random-Scan displays), Flat-Panel Displays. **Input devices:** Keyboards, Mouse, Joysticks, Digitizers, Touch Panels and Light Pens. **Hard-Copy Devices:** Printers.

Unit 2: Geometric Transformations: (8)

2-D transformations: Basic Transformations: Translation, Rotation and Scaling.
Other Transformations: Reflection and Shearing. **3-D transformations:** Translation, Rotation (Coordinate-Axes and General 3D) and Scaling. Other Transformations: Reflection and Shearing

Unit 3: Windowing and Clipping: (8)

The Viewing Pipeline, Window-to-Viewport Coordinate Transformation, Clipping Operations, Point Clipping, **Line Clipping:** Cohen –Sutherland, **Polygon clipping:** Sutherland- Hodgeman. **Filled-Area Primitives:** Scan-Line Polygon Fill Algorithm, Inside-Outside Tests, Boundary-Fill Algorithm, Flood-Fill Algorithm.

Unit 4: Curves and Surfaces: (8)

Curved Lines and Surfaces, Quadric Surfaces, Spline Representations, Bezier Curves and Surfaces, B-spline Curves and Surfaces.

Unit 5: Introduction to OpenGL & GLUT Libraries: (6)

Introduction to OpenGL, OpenGL basic graphics primitives: The OpenGL data types, OpenGL state, establishing the coordinate systems, Line drawing in OpenGL, drawingpoly-lines and polygons, Design & use of GLUT & GLUI menus.

Unit 6: Computer Animation: (4)

Design of Animation Sequences, General Computer-Animation Functions, Raster Animations, Computer-Animation Languages, Key-Frame Systems.
Introduction to Interactive Graphics & usage of application tools of computer graphics (Blender, 3D studio, or Similar open source tools)

Text Books:
Computer Graphics C Version- 2nd Ed.- Donald D. Hearn, M. Pauline Baker (Pearson Education)
(For Unit 1, 2, 3, 4 and 6)

Reference Books:
1) Mathematical Elements for Computer Graphics - David F. Rogers, J. Alan Adams (MGH Int.)
2) Procedural Elements for Computer Graphics - David F. Rogers (MGH Int.)
3) Computer Graphics Using OpenGL -F.S. Hill Jr. Stephen M. Kelley, (Pearson Education)

Open Source Resources:
1) http://www.glprogramming.com/red/chapter01.html
2) http://courses.cs.vt.edu/~cs4204/lectures/opengl_basics.pdf
3) http://sse.tongji.edu.cn/linzhang/ACG12/opengl.pdf
4) http://www.inf.ed.ac.uk/teaching/courses/cg/Web/intro_ogl.pdf

Term Work: It should consist of minimum of 8-10 experiments based on the following topics: -
1) Installation of computer graphics devices and adapters.
2) 2D Transformations
3) 3D Transformations
4) Filling algorithms
5) Clipping / Windowing / Viewporting
6) Bezier curves
7) B-Spline curves
8) Construction of simple pictures by drawing line, polylines, polygons using openGL.
9) File format conversion (like Bitmap, PCX)
10) Animation (Moving of object)

2. INFORMATION SECURITY

Lectures: 3 hrs / week **Theory: 100 marks**
Tutorial: 1 hr/Week **Term work: 25 marks**

Prerequisites: Basic knowledge of Computer Networks, OSI layer ,TCP/IP model.

Course Objectives:

1. To understand basics of cryptography, how it has evolved, and some key encryption techniques.
2. To understand principal concepts, major issues, technologies, and basic approaches in information security.
3. To learn security policies such as authentication, integrity and confidentiality.
4. To understanding of major information security threats and countermeasures.

Unit 1: Overview and Classical Encryption Techniques
Overview: (2)
Computer Security Concepts, The OSI Security Architecture, Security Attacks, Security Services, Security Mechanisms, A Model for Network Security
Classical Encryption Techniques: (2)
Symmetric Cipher Model, Substitution Techniques, Transposition Techniques, Rotor Machines

Unit 2: Block Ciphers and the Data Encryption Standard (4)
Block Cipher Principles, The Data Encryption Standard (DES), A DES Example, The Strength of DES, Differential and Linear Cryptanalysis, Block Cipher Design Principles

Unit 3: Public Key Cryptography (4)
Principles of Public-Key Cryptosystems, The RSA Algorithm, Diffie-Hellman Key Exchange, ElGamal Cryptosystem

Unit 4: Cryptographic Data Integrity Algorithms
Cryptographic Hash Functions: (4)
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 Codes: (4)
Message Authentication Requirements, Message Authentication Functions, Message Authentication Codes, and Security of MACs, MACs Based on Hash Functions: MAC, MACs Based on Block Ciphers: DAA and CMAC

Digital Signatures: (3)
Digital Signatures, ElGamal Digital Signature Scheme, Schnorr Digital Signature Scheme, Digital Signature Standard (DSS)

Unit 5: Key Management and Distribution (4)
Symmetric Key Distribution Using Symmetric Encryption, Symmetric Key Distribution Using Asymmetric Encryption, Distribution of Public Keys, X.509 Certificates, Public Key Infrastructure

Unit 6: Network And Internet Security
Transport-Level Security (4)
Web Security Issues, Secure Sockets Layer (SSL), Transport Layer Security (TLS), HTTPS
Electronic Mail Security (3)
Pretty Good Privacy (PGP), S/MIME
IP Security (2)
IP Security Overview, IP Security Policy, Encapsulating Security Payload

Text Book:

1. Williams Stallings – Cryptography and Network security principles and practices. Pearson Education (LPE), Fifth Edition
2. Cyber Security ,Nina Godbole , Wiley Publications.
3. Cryptography & Network Security B.A. Forouzan McGrawHill.

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

Term work: It should consist of 8-10 assignments based on above topics and solving exercise problems mentioned in the text book.

3. INTERNET TECHNOLOGY

Lectures: 3 Hrs / Week	**Practical: 2 Hrs / Week**	**TermWork:25 Marks**
Theory: 50 Marks	**Online: 50 Marks**	**POE:50 Marks**

Prerequisites: Knowledge of Computer Networks.

Course Objectives:

1) To make students able to identify client-server model and implement it using Java Socket programming
2) To make students able to use and analyze various Protocols using Protocol analyzing tools like wireshark and tcpdump.
3) To introduce students with emerging protocols IPv6 and the ICMPv6 and write applications to communicate using IPv6
4) To make students familiar with architecture of WWW
5) To make students able to understand working of email system and write an application to send and receive e-mail
6) To make students able to identify various protocols related multimedia over Internet

Unit 1: Client-Server paradigm, Socket Interfaces, Protocol analyzing (7)

The client-server model and software design, concurrent processing in client-server software, algorithms and issues in client-server design, multi-protocol servers, multi-service servers, concurrency in clients, Unix Internet Super Server (inetd).The Socket Interface, Socket Java API: connection oriented- Socket and ServerSocket. Connectionless- DatagramSocket and DatagramPacket. Utilitity classes- URL, URLConnection, InetAddress, InterfaceAddress. tcpdump, wireshark.

Unit 2: IPv6 and ICMPv6 (6)

IPv6 addressing, IPv6 Packet format, Transition from IPv4 to IPv6, ICMPv6

Unit 3: DHCP, DNS, TELNET and SSH (8)

DHCP: Introduction, Previous Protocols, DHCP operation, Packet Format, DHCP Configuration.

DNS: Need, Name Space, Domain Name Space, Distribution of name space, and DNS in internet, Resolution, DNS massages, Types of records, Compression examples, encapsulation.

TELNET and SSH: Concept, NVT, Embedding, Options & options/sub-option negotiation, controlling the server, Out-of-band signaling, Escape charter, Mode of operation, user interface, security issue in telnet, **SSH**, format of SSH packets.

Unit 4: FTP, TFTP and HTTP **(6)**

FTP: Connections, Communication, Command processing, File transfer, User interface, Anonymous FTP, **TFTP. HTTP:** Architecture, Web Documents, HTTP Transaction, Request & Response messages: header & examples, Persistent vs. non persistent HTTP, Proxy Servers.

Unit 5: Electronic Mail and SNMP **(6)**

Architecture, User agents, addresses, delayed delivery, Aliases, Mail transfer agent SMTP commands & responses, mail transfer phases, MIME, Mail Delivery, mail access protocols, SNMP.

Unit 6: Multimedia in Internet **(6)**

Streaming stored audio/video, streaming live audio/video, real-time interactive audio/video, real-time transport protocol (RTP), real-time transport control protocol (RTCP), voice over IP (VoIP): session initiation protocol (SIP) and H.323

Textbooks:

1. TCP/IP Protocol Suite Edition 4 by Behrouz Forouzan (McGraw Hill)

Reference Books:

1. Internet and Web Technologies ,Raj Kamal McGraw Hill

Open Source Resources:

1. http://docs.oracle.com/javase/7/docs/api/java/net/package-summary.html
2. http://nmap.org/ncat/guide/

Lab work:

It should consist of 10 to 12 assignments to be implemented in JAVA (preferably on Linux). Following is the minimum list of practical problems.

1. Client program using TCP and UDP to connect to well known services. (ECHO, TIME OF DAY, FINGER, TIME, etc)
2. Study of tcpdump and wireshark.
3. Iterative UDP client-server application. Server should keep log of client requests.
4. Concurrent TCP client-server application. Server should keep log of client requests.
5. Client-Server application using IPv6.
6. Study of DNS client utilities and implementation of nslookup/host.
7. Implement trivial file transfer protocol using TFTP messages.
8. Implement simple web server. Use browser as a client for your server.
9. Send/receive mail using SMTP/POP3(IMAP) commands.
10. Develop personal web site using database connectivity.

4. OPERATING SYSTEM II

Lecture: 3 Hrs/Week **Practical: 2 Hrs/week**
Theory : 100 Marks **Term Work: 25 Marks**

Prerequisites : Basic fundamental knowledge of Operating System.

Course Objectives:

1) To understand fundamental concepts of Unix.
2) To study Buffer cache
3) To study File system in Unix & system calls
4) To study structure of process
5) To study Process control and scheduling
6) To study Memory management and I/O subsystem

Unit 1

Overview of the UNIX System (6)

System structure, user perspective, Operating System services, assumption about H/W. Architecture of UNIX operating system, introduction to system concepts, kernel data structure, system administration

Unit 2

The Buffer Cache (6)

Buffer headers, structure of the buffer pool, scenarios for retrieval of a buffer, reading and writing disk blocks, advantages and disadvantages of cache

Unit 3

Internal Representation of Files (12)

Inodes, structure of the regular file, directories, conversion of a pathname to inode, super block, inode assignment to a new file, allocation of disk blocks, other file types System Calls for the File System Open, Read, Write, Close, File Creation, and Creation of special files, change directory and change Root, Pipes, Mounting and Unmounting File Systems, Link, Unlink

Unit 4

The Structure of process (6)

Process stages and transitions, layout of system memory, the context of a process, saving Context of a process, manipulation of the process address space

Unit 5

Process Control & Scheduling: (6)

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. Process Scheduling, system call for time, clock

Unit 6

Memory Management and I/O Subsystem: (4)

Swapping, Demand paging, A Hybrid system with swapping and demand paging, Driver Interfaces, Disk Drivers, Streams

Text Book

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

Term Work:

It should consist of 10 experiments of implementation based on UNIX/LINUX operating system.
Proposed List of Experiment:

1) Study of Unix Operating System
2) Study & Implementation of General Utilities, Directory & File Utilities
3) Study & Implementation of pipes
4) Implementation of Scheduling Algorithms.
5) Study & Implementation of process related utilities
6) Study & implementation of Shell programming
7) Study of system startup & init
8) Semaphore implementation
9) Implementation of IPC using message
10) Implementation of IPC using shared memory
11) Memory allocation algorithm (best-fit, first-fit, worst-fit).

5. SOFTWARE TESTING AND QUALITY ASSURANCE

Lecture: 3 Hrs/week **Theory: 100 Marks**

Prerequisites Basic knowledge of Software Engineering

Course Objectives:

1) To provide knowledge about fundamentals of software testing and software quality
2) To understand the fundamentals of software verification
3) To understand and evaluate metrics and models used in software testing
4) To understand and compare testing web applications and desktop applications
5) To understand, compare and Choose from various software project assessment methods

Unit 1: **(4)**

Introduction: Some Software Failures, Testing Process, Some Terminologies, Limitations of Testing, The V Shaped software life cycle model

Unit 2: **(8)**

Software Verification: Verification Methods, SRS document verification, SDD document verification, Source code reviews, User documentation verification, Software project audit

Creating test cases from SRS and Use cases: Use Case Diagram and Use Cases, Generation of test cases from use cases, Guidelines for generating validity checks, strategies for data validity, Database testing

Unit 3: **(7)**

Regression Testing: What is regression testing?, Regression Test cases selection, Reducing the number of test cases, Risk analysis, Code coverage prioritization techniques

Object oriented testing: What is Object orientation?, What is object oriented testing?, Path testing, State based testing, Class testing

Unit 4: **(6)**

Measurement- what is it and why do it?: Measurement in everyday life, Measurement in software engineering, scope of software metrics

Metrics and Models in Software testing: Software Metrics, Categories of Metrics, Object oriented Metrics used in testing, what should we measure during testing, Software Quality attributes prediction models

Unit 5: **(7)**

Measuring Internal Product Attribute Size: Aspects of software size, Length, Reuse, Functionality

Measuring External product Attributes: Modeling software quality, measuring aspects of software quality

Unit 6: **(8)**

Testing Web applications: What is web testing?, functional testing, UI testing, Usability testing, configurations and compatibility testing, security testing, performance testing, database testing, post deployment testing, web metrics.

Automated Test data generation: Automated Test Data generation, Approaches to test data generation, Test data generation tools

Text Books:

1) Software testing: Yogesh Singh, Cambridge University Press, First Edition

2) Software Metrics – A Rigorous & Practical approach: Norman Fenton, Shari Lawrence Pfleeger, 2nd Edition (Thomson Press) (for unit 4 Measurement-what is it and why do it? and unit 5)

3) Software Quality Engineering , Jeff Tian , Wiley India Ltd.

Reference Books:

1) Foundations of Software testing: Aditya P. Mathur, Pearson, Second Edition

2) Software Testing: Ron Patton, Pearson (SAMS), Second Edition

3) Software Quality , Mordechai Ben Menachem, Garry S. Marliss ,BS Publications

6. APPLICATION DEVELOPMENT TOOL – II

Lecture: 2 Hrs/week **Practical: 4 Hrs/week**
Term Work: 50 Marks **POE: 50 Marks**

Prerequisites: Basic knowledge of Object Oriented Programming

Course Objectives:

1. To understand fundamentals of .NET framework
2. To understand fundamental concepts like objects, classes, interfaces, polymorphism, delegates and events and its implementation in C#
3. to understand robust GUI applications using event handling and Windows form controls with proper exception handling
4. To understand I/O concepts, database connectivity using ADO.NET and collections and generics in C#
5. To analyse problems and devise suitable solution using C#.NET features

Unit 1: **(4)**
Introduction to .net: Evolution of .net, Benefits of .net, CLR, CTS, MSIL, JIT, BCL, metadata and assemblies in detail, GAC and strong name assemblies, Security Manager

Unit 2: **(3)**
C# fundamentals: Data types - Value types, Reference types, boxing and unboxing, Arrays, Pass by value and by reference and out parameters, params parameter
Namespaces, classes, objects, structs: definition and creation

Unit 3: **(8)**
Delegates and Events: Creating and using delegates, multicasting with delegates, event sources, event handlers
GUI Programming: Introduction to GUI Application and their components,
Windows forms – buttons, check boxes, radio buttons, panels, group boxes, list boxes, picture boxes, Menus, ToolStrips, StatusStrips and progress bars, events, Creating and using MDI application

Unit 4: **(4)**
File handling: The abstract stream class, working with StreamWriters and StreamReaders, Working with StringWriters and StringReaders, Working with BinaryWriters and BinaryReaders

Unit 5: **(4)**

ADO.NET: Exploring ADO.net Entity framework, Connected and disconnected architecture, data access with ADO.net

Unit 6: **(3)**

Collection and Generic: Collection classes in .net, Understanding Generics, generic collection classes in .net

Text Book

1. C# 4.0 The Complete Reference : Herbert Schildt, McGraw Hill

Reference Books:

1 Microsoft Visual C# 2010 Step by Step: John sharp, Microsoft Press
2 .NET 4.5 Programming (6 – in -1) Black Book – Kogent – Dreamtech Press
3 CLR via C# :Jeffrey Richter, Microsoft Press, 3rd edition
4 ASP.Net 4.5 Black Book ,Dreamtech ,Wiley International.

Guidelines for term work distribution

The distribution of the term work marks is as follows

- 25 marks for performance in practical and experiments
- 25 marks for mini-project to be developed using C#.

Guidelines for experiment list

The experiment list should consist of minimum 14 practical assignments on the above topics. Each experiment should be a problem statement which can be solved using some features of .Net and C#. Sample Experiment List is given below

1) Introduction to .Net framework & implementation of simple console application.
2) Study and implementation of different types of Constructors in C#.
3) Write a program to study use of Properties in C#.
4) Write a program to implement inheritance concepts.
5) Program to implement Different types of Delegates.
6) Program to demonstrate the events handler in C#.
7) Study of Collections and Generics and create a simple shopping cart Application that can be sorted by the price of the items using ArrayList.
8) Study of window-based application.
9) Program to study various controls for windows form application.
10) Create a small registration form layout using Windows Form Applications.
11) Demonstrate the Menu controls and Different Dialog controls in windows form application.
12) Study and Implementation of File Handling.
13) Program to display the account details with help of ADO.Net and windows form application.

14) Demonstrating dataset, data adapter and grid views in disconnected data access layer of ADO.NET.
15) Introduction to Visual web development application with ASP.NET.

Guidelines for Mini-project

Three students (Maximum) in a group will carry out a mini project.

A batch of practical should be divided into mini project groups. The faculty should guide the project group for selection of the topic and the work to be done. The mini project should consist of defining the problem, analyzing, designing the solution and implementing it using .NET. The faculty shall monitor the progress periodically. A presentation based on the above work is to be given by the group at the end of the semester. The work will be jointly assessed by a team of faculty from the department. A hard copy of project report, along with a softcopy of the programs is to be submitted to the department.

7. Seminar

Tutorial: 1 hr/Week **Term work: 50 marks**

Students should deliver seminar individually. It should consist of a talk of 30-45 minutes on a topic preferably from the area in which a student intends to work for his Project in B.E Semester – VII and Semester – VIII or any upcoming technology not covered in syllabus.

The seminar to be delivered by students should be assessed by a panel of at least two faculties within the department.
The assessment for the seminar should include but not limited to following points.

1) Novelty of the topic
2) Technical depth
3) Organization of the topic
4) Presentation skills
5) Communication skills
6) Answering Questions raised by faculties

The seminar report there-on is to be submitted which is to be internally assessed for 50 marks.

SHIVAJI UNIVERSITY, KOLHAPUR

(To be implemented from Academic Year 2015-16)

Subject Equivalence

Class : TE Part I (Semester V) **Branch : Information Technology**

Name of Subject (old)	New
1. Operating System-I	Operating Systems -I
2. Computer Networks	Computer Algorithm
3. Object Oriented Modeling & Design	Object Oriented Modeling & Design
4. System Software	System Programming
5.Organisational Management & Behaviour	Computer Graphics
6. Application Development Tool I	Application development Tool 1
7. Mini project	Soft Skills
Total	

Class : TE Part II (Semester VI)

Name of Subject (old)	New
1 Data Base Engineering	Data Base Engineering
2. Operating System II	Operating System II
3. Internet Technology	Internet Technology
4. Information System Security	Information Security
5. Software Testing & Quality Assurance	Software Testing & Quality Assurance
6. Application Development Tool II	Application Development Tool II