SHIVAJI UNIVERSITY, KOLHAPUR

REVISED STRUCTURE AND SYLLABII OF T.E. PART – I & II OF COMPUTER SCIENCE AND ENGINEERING .

T. E. PART – I

Sr. No.	Subject	L	T	P	Theory Marks	T/W	POE	Total Marks
1.	Advanced Microprocessors.	4	--	2	100	25	--	125
2.	Computer Graphics	3	--	2	100	25	--	125
3.	System Programming	3	--	2	100	25	50	175
4.	Operating Systems - I	3	1	--	100	25	--	125
5.	Computer Algorithm	3	1	--	100	25	--	125
6.	Programming Laboratory – III	2	--	4	---	25	50	75
	Total	18	2	10	500	150	100	750

T. E. PART – II

Sr. No.	Subject	L	T	P	Theory Marks	T/W	POE	Total Marks
1.	Compiler Construction	4	--	2	100	25	--	125
2.	Operating Systems - II	4	--	2	100	25	--	125
3.	Database Engineering	4	--	2	100	25	50	175
4.	Software Engineering	4	1	--	100	25	--	125
5.	Programming Laboratory - IV	2	1	2	---	25	50	75
6.	Seminar	--	--	2	---	25	--	25
	Total	18	2	10	400	150	100	650

Note : 1. The term work as prescribed in the syllabus is to be periodically and jointly assessed by a team of teachers from the concerned department.

2. In case of tutorials, students of different batches be assigned problems of different types and be guided for the solution of the problem during tutorial session. Problems thus solved be translated into computer programs wherever applicable and executed by respective batches during practical session.

3. The assignments of tutorials and practicals need to be submitted in the form of printout and /or written journal.

4. Breakup of termwork marks shall be as follows:

a) Tutorial assignments and/or practical performance – 15 marks.
b) Mid-semester test – 3 marks
c) End-semester test – 3 marks
d) Practical / tutorial attendance – 2 marks
e) Theory lecture attendance – 2 marks

5. Instructions for POE : Set of problems for POE should be based on the list of term work assignments, however they should not be exactly the same.

T.E. (Computer Science & Engg.) Part – I (Revised)

1. ADVANCED MICROPROCESSORS

Lectures : 4 hrs/week — Theory : 100 marks
Practicals : 2 hrs/week — T/W : 25 marks

SECTION – I

1. 8086 Architecture : Detail study of 8086 architecture, addressing modes, instruction formats, data transfer instructions, string instructions, logical instructions, arithmetic instructions, processor control instructions, comparison of 8086 with 8088, assembly language programming. -10

2. 8086 CPU Module: Basic 8086 CPU design, generating system clock and reset signals, microcomputer bus type and buffering techniques. - 5

3. System Bus Structure: Basic 8086 configurations, maximum and minimum mode, system bus timing, interrupts and interrupt responses. - 3

4. 8087 NDP Module: Data types, processor architecture, instruction set, programming 8087. - 6

SECTION – II

5. 80386 Architecture : Architecture and signal descriptors, Register organization, Addressing modes, Extended instruction set. - 6

6. Real mode operation of 80836 : Real mode operation, Memory addressing and interrupt processing. - 5

7. Protected mode operation of 80386 : Protected mode operation, memory organization – segmentation, descriptor types, paging, interrupt processing in protected mode. - 6

8. 80386 Memory Management Unit : MMU, virtual memory, descriptor tables GDT, LDT, IDT. - 5

Books :

1. The 8086/8088 family design, programming and interfacing – John F. Uffenbeck (PHI)
2. Intel 8086, 80386 manuals.
3. Advanced 80386 programming – Turley (MGH).

Reference:

1. Microprocessors, Interfacing & Assembly language programming – Douglas Hall (TMGH).
2. Microcomputer system – The 8086/8088 family – Liu & Gibson (PHI).

Term work : It shall consist of minimum 8-10 experiments based on the following topics

1. Assembly language programming for 8086.
   - Study of instruction set, Use of MUL/DIV instructions, Use of string processing instruction, use of XLAT instruction for code conversion.
2. Assembly language programming for 8086/8087
   - Study of NDP instruction set, Use of floating point ADD/SUB/MUL/DIV instructions, Use of F.P. instruction for generating Sine/Cosine/Exp/Log functions.
3. Use of ROM-BIOS services
4. Use of DOS interrupt services.
5. Programs based on 386 addressing modes.
6. Programs based on bit manipulation instructions using assembly language or C.
7. Programs to find square-root of 16-bit number.

T.E. (Computer Science & Engg.) Part – I (Revised)
2. COMPUTER GRAPHICS

Lectures : 3 hrs/week — Theory : 100 marks
Practicals : 2 hrs/week — T/W : 25 marks

SECTION - I

1. Introduction to graphics devices:
Picture representation, display devices , display adapters , Types of printers , Plotters & input devices., LCD, AGP. 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.
7

3. Raster Scan Graphics:
Bresenham's line and circle drawing algorithms, scan conversion, RLE, Frame buffer, Scan converting polygons - Edge fill and Seed fill algorithms, Antialiasing and Halftoning. 6

SECTION - II

4. Clipping and Display file Compilation :
Sutherland - Cohen line clipping algorithm, Windowing and Viewporting, Segmented display file, structure and compilation. 4

5. Hidden Surface and hidden Line Removal :
Backface removal algorithm, Z- buffer, Warnock algorithm, Hidden line elimination. 4

6. Plane Curves and Space Curves :
Curve Representation, Non-parametric and parametric curves, representation of space curves, Cubic Spline, Parabolic Blended curves, Bezier curves and B-spline curves. 9

Text Book :

1. Mathematical elements for Computer Graphics - David F. Rogers, J. Alan Adams (MGH Int.) (For chapters 1,2,6)
2. Procedural elements for Computer Graphics - David F. Rogers (MGH Int.)
(For chapters 3,4,5)

References :

1. Principles of Interactive Computer Graphics - Newman Sproul (MGH)
(For chapters 1,4)
2. Computer Graphics – Hearn & Baker.
3. Fundamentals of Computer Graphics - Steven Harrington (MGH).
4. Computer Graphics (second Edition) - Zhigang Xiang & Roy Plastock (Schaum's Outline Series, TMGH)

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. Bresenham's Line/Circle generation algorithm.
5. Filling algorithms.
6. Clipping, / Windowing, / Viewporting.
7. Hidden line/surface elimination algorithms.
8. Cubic Spline / Parabolic Blending curves.
9. Bezier / B-Spline curves.
10. File format conversion (like Bitmap, PCX)
11. Animation (Moving of object)

T.E. (Computer Science & Engg.) Part – I (Revised)
3. SYSTEM PROGRAMMING

Lecture : 3 Hrs/Week
Practicals : 2 hrs/week

Theory : 100 marks
T/W : 25 marks
POE : 50 marks.

SECTION - I

1. Language Processors**:** Introduction, language processing activities, Fundamentals of language processing, Fundamentals of language, Specification, language Processor development tools. 4

2. Assemblers: Elements of assembly language programming, A simple assembly scheme, Pass structure of assemblers, design of a two pass assembler, A single pass assembler for IBM PC. 4

3. Macros and Macro Processors: Macro definition and call, Macro Expansion, Nested macro calls, Advanced macro facilities, Design of macro preprocessor. 8

SECTION - II

4. Compilers and Interpreters: Aspects of compilation, memory allocation, compilation of expressions, compilation of control structures, code optimization, Interpreters. 7

5. Linkers : Relocation and linking concepts, design of a linker, Self-relocating programs, A linker for MS DOS, Linking for overlays, Loaders. 7

6. Software tools : Software tools for program development, Editors, Debug monitors, Programming Environments, User interfaces, DLLs. 4

Text books :
1. System Programming and operating systems – 2nd Edition D.M. Dhamdhere (TMGH)

Reference books :
1. .System Programming -- J. J. Donovan (Mc-Graw Hill)

Practicals : Minimum of 8-10 practical assignments should be carried on based on –

1. Implementation of Macros.
2. Implementation of Nested macros.
3. Design and implementation of 1 pass assemblers.
4. Design and implementation of 2 pass assemblers.
5. Symbol table generation for input *.c file.
6. Design Lex specifications for the tokens – keywords, identifiers, numbers, operators, white spaces.
7. Implementation of Toy-code generator.
8. Simulation of linkers.
9. Simulation of loaders.
10. 3-4 assignments on DLL on Linux shared library.
11. Study of different debugger tools.

T.E. (Computer Science & Engg.) Part – I (Revised)
4. OPERATING SYSTEM - I

Lectures : 3 Hrs/Week Theory :100 Marks
Tutorial : 1 Hr/week T/W : 25 marks

SECTION - I

1. Introduction : What is an operating system? , Simple Batch System, Multiprogrammed Batch System ,Time Sharing System, Personal Computer System, Parallel System, Real Time System, System Calls. 4

2. Process : Process Concept ,Process Scheduling, Operation on process ,Cooperating process , Threads , Interprocess Communication(Algorithms evaluation). 6

3. Process Scheduling: Basic concept ,Scheduling Criteria , Scheduling Algorithms, Multiple processor scheduling , Real time scheduling. 4

4 Interprocess synchronization: Background , Classical problems of synchronization , Critical Region , The critical section problem , Synchronization Hardware Monitors , Semaphores. 5

SECTION - II

5. Deadlocks: System modes ,Deadlock characterization , Methods for handling deadlocks Deadlock prevention , Deadlock avoidance , Deadlock detection Recovery from deadlock , Combined approach to dead lock. 7

6. Memory management: Background, Logical Versus Physical Address space, Swapping Contiguous Allocation , Paging, Segmentation , Segmentation with paging. 4

7. Virtual Memory: Background , Demand paging , Page replacement , Page replacement algorithms , Allocation of frames ,Thrashing(Only concept), Demand segmentation. 5

8. I/O system : Overview , I/O hardware ,Application I/O interface , Kernel I/O subsystem, Transforming I/O request to hardware operation. 4

Text Book :

1. Operating System concepts – 5th Edition – Silberschatz Galvin (John Wiley).

Reference:

1. Operating system A design oriented approach – Charles Crowley (TMGH)
2. Operating system with case studies in Unix, Netware and Windows NT – Achyut S. Godbole (TMGH).

Tutorial:

1. Students of different batches be assignment different exercise problems of the books mentioned and should be guided for their solution.
2. Case study of different Operating systems – DOS, Unix/Linux, Windows.

T.E. (Computer Science & Engg.) Part – I (Revised)
5. COMPUTER ALGORITHMS

Lectures : 3 hrs/week
Tutorials: 1 hr/week

Theory : 100 marks
T/W : 25 marks

Note : The scope of the subject is as per text book. A complete analytical treatment is expected with an emphasis on complexity analysis rather than algorithms.

SECTION - I

1. Introductiion : What is algorithm, Algorithm Specification, Recurrance relations, Performance Analysis. 5

2. Divide and Conquer : The general method, Binary search, Finding the maximum and minimum, Mergesort, Quicksort, Selection sort and analysis of these algorithms. 7

3. The Greedy method : 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. 7

SECTION - II

4. Dynamic Programming : The 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. Basic Traversal and Search Techniques : Techniques for Binary Trees, Game Tree; Techniques for Graphs – Breadth First Search & Traversal, Depth First Search & Traversal, AND/OR graphs; Connected components and Spanning Trees; Bi-connected components and depth first search . 7

6. Backtracking : The general method, 8-queen problem, sum of subsets, Knapscak Problem, Hamiltanion Cycle , Graph Coloring. 6

Text Book :

1. Fundamentals of Computer Algorithms – Horowitz, Sahni & Rajasekaran (Galgotia Publications)

References :

1. Fundamentals of Computer Algorithms - Horowitz and Sahni (Galgotia Publishers)
2. Design and analysis of algorithms - Aho, Hopfcraft and Ullman (Addison wesley).
3. Introduction to Algorithms – Thomas Cormen (PHI Publication)
4. Introduction to Design and Analysis of Algorithm – By Goodman (McGrawhill)

Tutorials : Students of different batches be

a) given different algorithms and asked to analyze for their performances. (find space & time complexities).
b) Assigned different exercise problems of the books mentioned in the syllabus including chapter 2 of the text book and be guided for the solution of the problems during tutorial session.

T.E. (Computer Science & Engg.) Part – I (Revised)
6. PROGRAMMING LABORATORY -III

Theory – 2 Hrs/ Week POE : 50 marks
Practical – 4Hrs / Week T/W : 25 marks

1. An Introduction to Java - Features of JAVA language, Java Virtual Machine and Java Programming Environment, Fundamental Programming Structures in Java , Interfaces and Inner Classes, static and non-static inner classes, Packages and access control mechanism, Comparison of Java with C++.

2. Error Handling and Exceptions, Debugging.

3. I/O programming – Hierarchy of classes in I/O Package, Streams: Character oriented and Byte oriented. Reading basic data types from keyboard. File handling in Java.

4. Event Handling in Java - Event delegation model (MVC model), Classes supporting event handling.

5. Multithreading – Classes supporting Thread creation, Thread States & Synchronization of threads. Thread groups. Deadlock handling.

6. GUI Design in Java – Hierarchy of classes in AWT package, User Interface Components with swings, Applets.

7. Network programming with java - Hierarchy of classes in NET package. Client server Programming , Concurrent and Iterative server design., RMI package.

8 Database programming in Java

9.Native code and security issues in java.

Text Books :

1. Core Java Fundamentals Vol -I (The Sun Microsystems Press Java Series) Cay S. Horstmann, Gary Cornell

2. Core Java Vol – II (The Sun Microsystems Press Java Series) Cay S. Horstmann, Gary Cornell

Reference:

1. Java 2 Complete Reference – 5th Edition – Herbert Schildt (TMGH).
2. How to Java – Dietel & Dietel.

T.E. (Computer Science & Engg.) Part – II (Revised)
1. COMPILER CONSTRUCTION

Lectures : 4 hrs/week — Theory : 100 marks
Practicals : 2 hrs/week — T/W : 25 marks

SECTION - I

1. Introduction to Compiling:
Compilers, Phases of a compiler, Compiler construction tools, A simple one pass compiler. - 4

2. Lexical Analysis: - 6
Role of a Lexical analyzer, input buffering, specification and recognition of tokens, finite automata implications, designing a lexical analyzer generator .

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:
Syntax directed definitions, construction of syntax tree, Bottom-up evaluation of S-attributed definitions, L-attributed definitions, Top-down translation and Bottom-up evaluation of inherited attributes, analysis of syntax directed definitions. - 7

SECTION - II

5. Run Time Environments :
Source language issues, storage organisation and allocation strategies, parameter passing, symbol table organisations and generations, dynamic storage allocations. - 5

6. Intermediate Code Generation :
Intermediate languages, declarations, assignment statements and boolean expressions, case statements, back patching, procedure calls. - 5

7. 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, assignment and basic blocks, code generation from Dags and the dynamic code generation algorithm. - 6

8. Code Optimization :
Sources of optimization, Peephole optimization and basic blocks, loops in flow graphs, Data flow analysis and equations, code improving transformation and aliases, Data flow analysis and algorithms, symbolic debugging of optimized code. - 6

Text Book :

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

References : -

1. Compiler Construction - Dhamdere (Mc-Millan)
2. Compilers - Principles, Techniques and Tools - A.V. Aho, R. Shethi and J.D. Ullman (Addison wesley publishing company.)
3. Compiler Construction - Barret, Bates, Couch (Galgotia)
4. Unix Programming - Pepkin Pike.

Term Work : It should consist of minimum 8-10 experiments based on the above topics. Following experiments MAY be conducted for the term work.

1. Generate a grammar for a language whose description is known.
2. Design a lexical analyser for a language whose grammar is known.
3. Implement a recognizer for the language in 2.
4. Implement a parser for the language given in 2.
5. Generate a symbol table for the language given in 2.
6. Generate 3 address code for the language given in 2.
7. Implement code optimization techniques on the code produced in 6.
8. Generate target code for the code optimized in 4, considering the target machine to be X86.
9. Use of LEX & YACC utilities.
10. Case study of Fort Language (IIT Bombay).

T.E. (Computer Science & Engg.) Part – II (Revised)
2. OPERATING SYSTEM – II

Lectures : 4 Hrs/Week — Theory : 100 Marks
Practical : 2 Hrs/ Week — Term work : 25 Marks

SECTION – I

1. Introduction : 07

General Overview of the System - History, System Structure, User Perspective, Operating System Services, Assumption About Hardware. Introduction to the KERNEL - Architecture of UNIX OS, Introduction to system concepts, Kernel Data Structure, System Administration.

2. The Buffer Cache : 5

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

3. Internal Representation of Files : 5

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.

4. System calls for the file System : 5

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 Owner and Change Mode, Stat and Fstat, Pipes, Dup, Mounting and Unmounting file systems, Link, Unlink, File System Abstractions,
File system maintenance.

SECTION – II

5. The Structure of process : 5

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. Process Control : 5

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. Process Scheduling and Time : 3

Process Scheduling, system call for time, clock.

8. Memory management policies : 4

Swapping, Demand passing, a hybrid system with demand paging and swapping

9. The I/O Subsystem : 5

Driver interfaces, disk drives, terminal drivers, Streams.

Text Book :

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

Reference:

1. Unix concepts and administration – 3rd Edition – Sumitabha Das (TMGH).

Term Work : It should consist of minimum 8-10 experiments based on the above topics.

T.E. (Computer Science & Engg.) Part – II (Revised)
3. DATABASE ENGINEERING

Lectures : 4 Hrs/Week
Practicals : 2 Hrs/Week

Theory :100 Marks
T.W. : 25 Marks
POE : 50 Marks

SECTION – I

1. Introduction: Purpose of Database Systems, Data abstraction, Data Models, Entities and Entity sets, Mapping Constraints, E-R Diagram, Reducing E-R Diagrams to Tables, Generalization, Aggregation. (4)

2. Relational Model: Structure of Relational Databases, The Relational Algebra, The Tuple Relational Calculus, The Domain Relational Calculus, Structured Query Language(SQL). (7)

3. Integrity Constraints and Design: Domain Constraints, Referential Integrity, Functional Dependencies, Normalization using Functional Dependencies, -canonical cover. (7)

4.File and System Structure: Overall System Architecture, File Organization, Organization of Records into Blocks, Sequential Files, Mapping Relational Data to Files, Data Dictionary Storage, Buffer Management. (4)

SECTION - II

5. Indexing and Hashing : Basic Concepts, Indexing, B+ Tree Index Files, B-Tree Index Files, Static Hash Functions, Dynamic Hash Functions, Comparison of Indexing and Hashing, Multiple Key Access. (5)

6. Query Processing : Query Interpretation, Equivalence of Expressions, Estimation of Query Processing Cost, Estimation of Costs of Access using Indices. (5)

7. Crash Recovery : Failure Classification, The storage Hierarchy, Transactions Model, Log-Based Recovery, Buffer Management, Checkpoints, Shadow Paging, Failure with Loss of Non-Volatile Storage, Stable Storage Implementation. (5)

8. Concurrency Control : Schedules, Testing for Serializability, Log-Based Protocols, Time-Stamp Based Protocols, Validation Techniques. (7)

Text Book :

1. DataBase System Concept by Henry F. Korth, Abraham Silberschatz, Sudarshan (McGraw Hill Inc.) Fourth Edition
2. DataBase System Concept by Henry F. Korth, Abraham Silberschatz, (McGraw Hill Inc.)

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 – Masri and Navathe (Bengamin Cummings, 1989).
4. Database design, application development & administration – Michael V. Mannino (MGH- International Edition).

Term Work :

1. It should consist of minimum 8 experiments based on above topics and should be implemented in C++/Java only. No DBMS package should be used for practical assignments.

2. Set of assignments is listed below:

   1. **Title : ER Diagrams & Normalization**
      Draw ER diagrams (around 10 in number) for college Student Activities & Convert them into tables. Apply normalization. Display constraints.
   2. **Title : Data Dictionary**
      Write program to create tables, along with constraints and store them in a file, which will work as DD for later assignments.
   3. **Title : Insert Data**
      Write program to Insert data in tables created in assignment 2. Store data in separate File / Table. Implement insert operation as transaction.
   4. **Title : Modify Data**
      Write program to modify data in tables, which is inserted in assignment 3. Implement modify operation as transaction.
   5. **Title : View Data**
      Write program to view table data. Accept table attribute for ordering dynamically.
   6. **Title : B+ Tree Indexing Technique**
      Write program to implement B+ Tree Index (n=3 or n = 5) on the data created until now.
   7. **Title : Dynamic Hashing Technique**
      Write program to implement Dynamic Hashing on the data created until now.
   8. **Title : Database Logs**
      Write program to create logs of the activities of assignment 3 & 4. Choose either Immediate Log OR Deferred Log.
   9. **Title : Concurrency Control**
      Write program to simulate any one concurrency control Protocol.
   10. **Title : Canonical cover & Closure**
      Given a set of functional dependencies Find canonical cover & closure.

T.E. (Computer Science & Engg.) Part – II (Revised)
4. SOFTWARE ENGINEERING

Lectures : 4 hrs/weel — Theory : 100 marks
Tutorials : 1 hr/week — T/W : 25 marks

SECTION-I

1. Introduction :
   The S/W problem , S/W engg. Problem , the S/W engg. Approach. (2)

2. Software Processes : Software Process, Characteristics of a software process, Software development process, project management process, Software configuration management process, process management process. (4)

3. S/W requirements analysis and specification , S/W requirements, problem analysis , Requirements Specification , validation, metrics. (3)

4. Planning a Software Project : (5)
   A] Cost estimation, project scheduling, staffing and personnel planning,
   B] Software Configuration Management plans, Quality Assurance plans.
   C] Project Monitoring Plans, Risk Management.

5. Function oriented design: Design principles, module level concepts, Design notation and specification Structured Design methodology, Verification, Metrics. (4)

6. The project management plan : Team management , customer communication and issue resolution, The structure of the project management plan. (3)

SECTION-II

7. Object Oriented Design. OO Analysis and OO Design, Concepts, Design Notation and specification, Design methodology, module specifications, Detailed design, Verification, Metrics. (4)

8. Coding: Programming Practice, verification, Metrics. (3)

9. Testing: Testing Fundamentals, Functional testing, Structural testing, Testing object oriented programs, Testing process Metrics-Reliability Estimation. (5)

10. Managing S/W projects: Processes and project management and the CMM , project management process, Training for project managers, SEPG support to projects. (5)

11. Project monitoring and control: Project tracking, milestone analysis, Activity – level Analysis using SPC, Defect Analysis and prevention, Process monitoring and Audit (3)

Text Books :

1) An integrated approach to S/W engineering,
Second edition - Pankaj Jalote. (Narosa Publishers)

References :

1) Software Project Management in practice – Pankaj Jalote.(Pearson Education)
2) Software Engineering. : Practitioner's Approach – Roger S. Pressman (TMGH)
3) Software Engineering - Jawadekar W.S. (TMGH)

Term work :

- Mini project may be assigned to group of 4 to 5 students in a batch. This group will carry out study of software life cycle for the selected mini project using appropriate UML tools.
- Case studies based on text book and reference books given in syllabus.

T.E. (Computer Science & Engg.) Part – II (Revised)
5. PROGRAMMING LABORATORY - IV

Lectures :2hrs/week
Practicals : 2hrs/week
Tutorial : 1 hr/week.

T/W :25 marks
POE : 50 marks.

1. Introduction to Windows Operating system. Developing window application in SDK.
2. Detail study of Windows messages.
3. Introduction to GDI –understanding DC, scrollbars, drawing lines, dots, GDI mapping modes, drawing filled areas.
4. Working with keyboard & mouse – understanding keyboard &mouse basics, keyboard messages, mouse messages, capturing mouse
5. Working with Window controls – study of various buttons, study of controls , edit box , scrollbars list box etc.
6. Working with menus – adding icons, cursor, custom resources ,adding menus to application, enabling disabling menu items.
7. Working with Dialog boxes – modal dialog box ,modeless dialog box, common dialog box
8. Working with clip board – standard data formats , memory allocation, transferring data to the clip board , getting data from clipboard.
9. Developing Multiple Document Interface (MDI)
10. Developing Dynamic Linked Libraries (DLL)
11. Introduction to MFC.
12. Study of Documents ,Views& frames- creating SDI application, understanding document template, using documents& views together

Text Books :

1. Programming Windows fifth edition by Charles Petzold Microsoft press
2. Programming Visual C++ by david Kruglinski,shepherd,Wingo Microsoft press
3. Complete Reference VC++ 6 – Pappas Murray (TMGH).

Reference Books :

1. Teach yourself Visual C++ in 21days by Chapman Techmedia publications
2. Practical Visual C++ by Jon Bates & Tim Tompkins (PHI)
3. MFC Programming from the ground up by Herbert Schildt (TMH)
4. Visual c++ Progrmming vol – I by Yashwant kanetkar
5. Visual c++ Progrmming vol – II by Yashwant kanetkar

Term Work: The term work should consists of minimum 8-10 experiments based on above mentioned topics .

Tutorials : Students of different batches be assigned different programming problems and be guided for the solution during the tutorial session. Problems thus solved be translated into computer programs by respective batches during practical session.

T.E. (Computer Science & Engg.) Part – II (Revised)
6. SEMINAR

Practicals : 2hrs/week Term work : 25 marks

The groups of students of strength 4-6 should be formed by the end of T.E.-I. The projects for the group should be finalized by the end of 1st month of T.E.-II. Seminar should consist of a presentation of about 30-40 minutes by every individual student. The seminar should be based on topics in the area in which the students have carried on the literature survey and will work for their selected project in the final year. A report on the seminar should be submitted to the department. Assessment should be jointly done by panel of teachers consisting of respective guide and other teachers from the department.

REVISED STRUCTURE & SYLLABII B.E. PART I & II
OF COMPUTER SCIENCE AND ENGINEERING

B. E. PART – I

Sr. No.	Subject	L	T	P	Theory Marks	T/W	Orals	POE	Total Marks
1.	Advanced Computer Arch.	4	1	--	100	25	--	--	125
2.	Distributed Systems	4	--	2	100	25	--	50	175
3.	Network Engineering	2	--	4	---	25	--	50	75
4.	Information Technology	3	--	2	100	25	--		125
5.	Elective – I	3	1	--	100	25	--	--	125
6.	Project	--	--	4	--	25	50	--	75
		16	2	12	400	150	50	100	700

B. E. PART – II

Sr. No.	Subject	L	T	P	Theory Marks	T/W	Orals	POE	Total Marks
1.	Advanced Database Systems	4	--	2	100	25	--	--	125
2.	Component Technology	4	--	2	100	25	--	--	125
3.	Information Security	3	1	--	100	25	--	--	125
4.	Web Technology	2	--	2	---	25	--	50	75
5.	Elective – II	3	1	--	100	25	--	--	125
6.	Project	--	--	6	---	50	75	--	125
		16	2	12	400	175	75	50	700

ELECTIVE – I

1) Information Retrieval
2) Object Oriented Modeling & Design
3) Digital Signal Processing

ELECTIVE - II

1) Mobile Computing
2) Image Processing & Pattern Recognition
3) Artificial Neural Networks & Genetic Algorithms.

Any other elective based on the current developments with prior sanction from the University Authorities.

Note :

1. For the project head, the practical batch shall consist of 09 (Nine) students each.

2. Laboratory Term work as prescribed in the syllabus is to be periodically and jointly assessed by a team of teachers as appointed by the Head of the Institution.

3. In case of Tutorials, students of different batches be assigned problems of different types and be guided for the solution of the problem during tutorial session. Problems thus solved should be translated into computer programs wherever applicable and executed by respective batches during practical session.

4. The assignments of tutorials and practicals need to be submitted in the form of printout and /or written journal.

5. Breakup of term work marks shall be as follows:

a) Tutorial assignments and/or practical performance – 15 marks.
b) Mid-semester test – 3 marks
c) End-semester test – 3 marks
d) Practical / tutorial attendance – 2 marks
e) Theory lecture attendance – 2 marks

6. Instructions for POE : Set of problems for POE should be based on the list of term work assignments, however they should not be exactly the same.

Revised Syllabus for B.E. (Information Technology)

Information Technology

T.E. (IT)-Part-I - Introduced from July 2004

Sr. No	Subject	Load / Week			Marking Scheme				
		L	Pr.	T	Paper	TW	POE	OE	Total
1	Computer Organization & Architecture	3	-	-	100	-	-	-	100
2	Operating System –I	4	2	-	100	25	50	-	175
3	Computer Networks-II	4	2	-	100	25	50	-	175
4	Software Engineering	4	-	-	100	-	-	-	100
5	Digital Communication	3	2	-	100	25	-	-	125
6	Application Development Tool-I	2	4	-	-	25	50	-	75
	Total	20	10	-	500	100	150	-	750

Information Technology

T.E. (IT)-Part-II

Sr. No.	Subject	Load / Week			Marking Scheme				
		L	Pr.	Tu.	Paper	TW	POE	OE	Total
1	Image processing	3	-	-	100	-	-	-	100
2	Operating System-II	4	2	-	100	25	-	25	150
3	Data Base Engineering	4	2	-	100	25	50	-	175
4	Organizational Management & Behavior	3	-	-	100	-	-	-	100
5	Internet Technology	4	4	-	100	25	50	-	175
6	Application Development Tools-II	2	2	-	-	-	-	25	25
7	Seminar	-	2	-	-	25	-	-	25
	TOTAL	20	10		500	100	100	50	750

B.E. Information Technology

PART - I

Name of Subject	L	P	Theory Paper	T/W	OE	POE	Total
1. Object Oriented Modeling and Design (OOMD)	4	2	100	25	50		175
2. Advanced Database System	4	2	100	25	-	50	175
3. Information system security	4	-	100	-	-	-	100
4. Network Engineering	2	4	-	25	50	-	75
5. Elective-I	4	-	100	-	-	-	100
6. Project I		4		25	50		75
	18	12	400	100	150	50	700

PART -II

Name of Subject	L	P	Theory Paper	T/W	OE	POE	Total
1. Software testing & quality assurance	4	2	100	25	50	--	175
2. IT Business Methodology	4	--	100	--	--	--	100
3. Mobile Technology	4	--	100	--	--	--	100
4. Web. Technology	2	4	--	25	--	50	75
5. Elective –II	4	--	100	--	--	--	100
6. Project II		6		50	100	--	150
Total	18	12	400	100	150	50	700

Electives-I

1) Embedded & Real time system
2) ANN
3) Discrete event system simulator & Modeling

Electives-II

1) Intelligent Systems
2) Multimedia storage & communication
3) GIS & Remote sensing