

Solapur University, Solapur
Structure of T.E.(Computer Science and Engineering) Part I & II
w.e.f. Academic Year 2009-10.

T.E.(Computer Science and Engineering) Part - I

Sr. No	Subject	Teaching Scheme				Examination Scheme				
		L	T	P	Total	TH	TW	POE	OE	Total
1	Computer Network - II	4	-	2	6	100	25	-	-	125
2	Computer Organization	4	-	-	4	100	-	-	-	100
3	System Programming	3	-	2	5	100	25	50	-	175
4	Operating System – I	3	-	2	5	100	25	-	-	125
5	Design & Analysis of Algorithm	4	-	-	4	100	-	-	-	100
6	Java Programming	2	-	4	6	-	25	50	-	75
	Total	20	-	10	30	500	100	100	-	700

T.E.(Computer Science and Engineering) Part – II

Sr. No	Subject	Teaching Scheme				Examination Scheme				
		L	T	P	Total	TH	T/W	POE	OE	Total
1	Data Base Engineering	4	-	2	6	100	25	50	--	175
2	Compiler Construction	3	-	2	5	100	25	-	--	125
3	Operating System – II	4	-	2	6	100	25	50	--	175
4	Software Engg.	4	-	-	4	100	-	-	--	100
5	Artificial Intelligence	3	-	-	3	100	-	-	--	100
6	Advanced Java	2	-	2	4	-	25	50	--	75
7	Seminar	-	-	2	2	-	50	-	--	50
	Total	20	-	10	30	500	150	150	--	800

Note:-

1. Vacational Training (to be evaluated at B.E. Part-I) of minimum 15 days should be completed in any vacation after S.E.Part-II but before B.E. Part-I & the report should be submitted in B.E. Part-I.
2. The batch size for the practical/tutorials be of 15 students. On forming the batches, if the strength of remaining students exceeds 7 students, then a new batch may be formed.

T.E. (Computer Science & Engineering) Part – I
1. Computer Network-II

Lecture: 4 hrs/week
Practical: 2 hrs/week

Theory: 100 Marks
Term work : 25 Marks

Section I

1:Transport Layer –

(10)

Overview of IP Protocol ,UDP:process to process communication, user datagram, UDP operations
Use of UDP ,TCP:Tcp services, TCP features, segment of TCP connection, flow control, error control, congestion control,TCP timers

2: Client server model & socket interface –.

(5)

Client server model, concurrency, processes, sockets, byte ordering, address transformation. Socket system calls, connectionless iterative server, UDP client server programs, and connection oriented concurrent server. TCP client server programs

3:Host configuration -

(7)

BOOTP:Operations and packet format,DHCP: Static and Dynamic addressing, allocation, manual and Automatic configuration, packet format,

Section II

4 : Domain Name System (DNS) :

(4)

Name Space, Domain Name Space, distribution of name space, DNS in internet,Resolution, DNS messages, types of records, WINES.

5 : TELNET, FTP –

(7)

Concept, NVT, embedding, out of band signaling, escape character, mode of operations. FTP and TFTP .

6 :SMTP and WWW–

(6)

Architecture, user, agents, MTA,pop3, HTTP: Architecture , protocol.

7:Network OS :

(5)

Windows 2000 / NT:- System Overview,Windows Networking Architecture,File System – FAT 16,FAT32,NTFS.

Text Book :

1. TCP/IP Protocol Suite : Behroz A. Forozen (Second Edition) (Chapter 2)
2. TCP/IP Protocol Suite : Behroz A. Forozen (Third Edition) (Chapter 1,3,4,5,6)
3. Networking : The Complete Reference : Craig Zacker (Chapter 7) (Tata McGraw –Hill)

Ref Book : 1. TCP/IP Vol 3. : Client Server Programming & Application – Comer

Termwork : Minimum 8-10 Experiments based on following guidelines.

List of assignments:-

1. Configuration of network- Assigning IP Adress,Subnetmask,Default Gateway & Testing Basic Connectivity.
2. Implementing Client-Server program using Iterative UDP server.
3. Implementing Client-Server program using Iterative TCP server.
4. Implementing Client-Server program using Concurrent TCP server.
5. Simulation of DHCP.
6. Simulation of DNS.
7. Simulation of FTP.
8. Implementation of Chatting Application.
9. Installation & Configuration of –Windows 2000/2003 or Linux.
10. Study of existing network (e.g. college) and design of any new network.

T.E. (Computer Science & Engineering) Part – I
2. COMPUTER ORGANISATION

Lectures: 4Hrs./Week

Theory: 100

Section I

- 1. Introduction to CO -- (02)**
Generations of computers
- 2. Processor Level design -- (08)**
Instruction Sets-Instruction format, Instruction types; Bus-Hierarchical Architecture, RISC,CISC ;Fixed point Arithmetic: Addition, subtraction, multiplication(Booths Algorithms), Division(Restoring & non restoring);Floating Point Arithmetic: Addition (Examples based on different Algorithms).
- 3. Control Unit Design -- (04)**
Hardwired Control Unit: Design Methods(Sequence counter);Multiplier Control Unit (Introduction), (Implementation of Multiplier in each case).
- 4. Memory Organization -- (06)**
Virtual memory: Memory Hierarchy, Main memory allocation, Segments & pages.

Section II

- 5.Memory Design -- (08)**
Replacement Policies; High Speed memories-Interleaved memories, Cache, Associative.
- 6. Parallel Processing -- (08)**
Uniprocessor and Multiprocessor parallelism; Types of uniprocessor parallelism; Basics of Pipelining & vector processing; Difference between pipelining and vector processors; Multiprocessor Architecture-tightly coupled & loosely coupled ,Examples of loosely coupled
- 7.Pipelined Architectures – (08)**
Linear, Nonlinear pipeline ,pipeline hazards, bubbles in pipeline;

Text Books:

- 1.Computer Architecture & organisation – J.P.Hayes(MGH)
- 2.Computer Organisation & Architecture Hanasey Paterson
3. Advanced Computer Architecture Kai Hwang(MGH)

Reference Book

- 1.Computer Organisation -Hamacher Zaky(MGH)
- 2.Designing for Performance William Stallings

T.E. (Computer Science & Engineering) Part – I
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:** (7)
Introduction, language processing activities, Fundamentals of language processing, Fundamentals of language, Specification, language Processor development tools.
- 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 IBM PC.
- 3.Macros and Macro Processors:** (4)
Macro definition and call, Macro Expansion, Nested macro calls.

SECTION - II

- 4. Compilers and Interpreters:** (7)
Aspects of compilation, compilation of expressions, code optimization.
- 5. Linkers :** (6)
Relocation and linking concepts, design of a linker, Self-relocating programs, Linking for overlays.
- 6. Loaders:** (5)
Function of loader, general loader scheme, Absolute loader, Relocating loader, Direct linking loader, Dynamic loading, Design of direct linking loader.

Text books :

1. System Programming and operating systems – 2nd Edition D.M. Dhamdhare (TMGH) (unit-1,2,3,4,5)
2. System Programming -- J. J. Donovan (Mc-Graw Hill) (unit-6)

Reference books :

1. System Software- An Introduction to Systems Programming- 3rd Edition- Leland L. Beck (Pearson Education)

Termwork :

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.

SECTION – I

1.Introduction :

(4)

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

2.Process :

(6)

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

3.Process Scheduling:

(4)

Basic concept ,Scheduling Criteria , Scheduling Algorithms, Multiple processor scheduling , Real time scheduling.

4 Interprocess synchronization:

(5)

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

SECTION – II

5.Deadlocks:

(7)

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

6. Memory management:

(4)

Background , Logical Versus Physical Address space , Swapping Contiguous Allocation , Paging, Segmentation , Segmentation with paging.

7. Virtual Memory:

(5)

Background , Demand paging , Page replacement , Page replacement algorithms , Allocation of frames ,Thrashing(Only concept), Demand segmentation.

8. I/O system :

(4)

Overview , I/O hardware ,Application I/O interface , Kernel I/O subsystem, Transforming I/O request to hardware operation.

Text Book :

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

Reference:

2. Operating Systems: Internals and Design Principles, 5th Edition by William Stallings (PHI)

3. Operating system with case studies in Unix, Netware and Windows NT – Achyut S. Godbole (TMGH).

Termwork :**Practical List**

It should consist of the following practicals

Based on C and linux

1. Study of Unix Operating System.
2. Implementation of system calls : fork (), join(), abort (), suspend()
3. Implementation of FCFS scheduling algorithm.
4. Implementation of SJF (preemptive & non preemptive)
5. Implementation of round robin (RR).
6. Implementation of priority scheduling algorithm.

Based on Java Multithreading

7. Implementation of Mutual Exclusion 1st algorithm.
8. Implementation of Mutual Exclusion 2nd algorithm.
9. Implementation of Mutual Exclusion 3rd algorithm.
10. Implementation of Mutual Exclusion using semaphore (wait & signal)
11. Implementation of producer consumer problem (Bounded buffer)
12. Implementation of producer consumer problem (Unbounded buffer)

T.E. (Computer Science & Engineering) Part – I
5. DESIGN & ANALYSIS OF ALGORITHM

Lectures: 4 hrs/week

Theory: 100 marks

SECTION – I

1. **Introduction :** (7)
What is algorithm, Algorithm Specification, Recurrence relations, Performance Analysis?
2. **Divide and Conquer :** (9)
The general method, Binary search, Finding the maximum and minimum, Merge sort, Quicksort, Selection sort and analysis of these algorithms.
3. **The Greedy method :** (9)
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.

SECTION - II

4. **Dynamic Programming :** (10)
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.
5. **Basic Traversal and Search Techniques :** (8)
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.
6. **Backtracking :** (9)
The general method, 8-queen problem, sum of subsets, Knapsack Problem, Hamilton Cycle, Graph Coloring.

Text Book:

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

References:

1. Fundamental of Algorithm. – Gilles Brassard, Paul Bratley (Pearson Publication)
2. Introduction to Algorithms – Thomas Cormen (Pearson Publication)
3. Introduction to Design and Analysis of Algorithm – By Goodman (McGrawhill)
4. Design and analysis of algorithms - Aho, Hopcraft and Ullman (Addison wesley)

1. **Introduction to Java:** (03)
Overview of Java, Features of Java Language, Java Road Map, Data types and operators, Strings, Vectors. Type wrapper, classes in Java.
2. **Object oriented programming with Java** (04)
Define class, method, properties, Understanding usage of packages, Creating user defined packages, Managing classes under packages, Access modifiers, this keyword, Working with Constructors
3. **Inheritance and Polymorphism :** (03)
Types of inheritance in java, Multiple inheritance using interface, Method overriding, use of super final keywords, Creating abstract classes, Interfaces & methods, Finalization and Garbage collection
4. **Exceptions and Error handling:** (03)
Introduction to Exception, Dealing with exceptions, try, catch blocks, finally block, throw and throws statement, Common Exceptions, The Throwable class, User defined Exception
5. **I/O Programming:** (03)
Hierarchy of classes in I/O package, Streams: Character oriented and Byte oriented, reading basic data types from keyboard, File handling in Java.
6. **Multithreading:** (03)
Java thread model, Thread creation using Thread class and Runnable Interface, Thread priorities , Thread Synchronization, Thread groups, deadlocks
7. **GUI Design in Java and Event handling** (06)
Hierarchy of classes in AWT package, Applet programming, Event handling on TextField, Buttons, Checkboxes, Lists, Menus and MenuBar, Layout managers, Key and Mouse Events
8. **Network programming with Java:** (04)
Networking fundamentals, Client server programming: InetAddress, URLs, Sockets, Datagram Socket, Introduction to RMI
9. **Java Utilities and file compression:** (02)
Data structures using Java:Set,Map,List,Tree, Jar files, Data compression using Java.util. ZIP package

Text Books:-

1. Programming with Java a primer – E. Balgurusamy (TMGH)
2. Object Oriented Programming through Java- E Radhakrishna (University Press)
3. The Java Programming Language, 3rd Edition – Kea Arnold, David Holmes, James Gosling, Prakash Goteti

Reference Books:-

1. The Java language specification
(E-Book: [http:// Java-sun.com/docs/books/jeles/downloads/langspec 3.0 pdf](http://Java-sun.com/docs/books/jeles/downloads/langspec 3.0 pdf))
2. The complete Reference, Java2 (5th edition) – Herbert Schedt et. a (Osborn)
3. Java for Professionals – B.M. Harwani (SPD)

Termwork :

Practicals :

1. Scope of Object in Program.
2. Console I/O
3. Recursion in JAVA
4. Constructor
5. Instance Variable
6. Program on StringTokenizer
7. Applet
8. Multithreading
9. File Copy program
10. GUI and Event Handling
11. Concurrency
12. Socket based programs

T.E. (Computer Science & Engineering) Part – II
1. DATABASE ENGINEERING

Lectures : 4 Hrs/Week
Practical : 2 Hrs/Week

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

SECTION – I

1. Introduction:

(06)

Purpose of Database Systems, View of data, Database Languages, Database Architectures, Database users and administrators, Entity Relationship models, Constraints, E-R Diagram, E-R Design issue, Weak Entity sets, Extended E-R features, Database design for Banking Enterprise, Reduction to relation schemas.

2. Relational Model :

(10)

Basic Structure , Relational algebra Operations : Fundamental, Additional and Extended , Null values, Modification of Database, Structured Query Language(SQL) The Tuple Relational Calculus, The Domain Relational Calculus.

3. Relational Database Design :

(08)

Features of good Relational Designs, Atomic Domains and First Normal Form, Decomposition using Functional dependencies, Third Normal Form Functional-dependency theory, Decomposition using functional dependencies, Decomposing using Multi valued dependencies.

SECTION - II

4. Indexing and Hashing :

(06)

Basic Concepts, Ordered Indices, B+ Tree Index Files, B Tree Index Files, Static Hashing, Dynamic Hashing, Comparison of Indexing and Hashing, Multiple Key Access.

5. Transactions and Concurrency Control :

(10)

Transaction concepts, Transaction state, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation , Testing for Serializability ,Log based protocols, Time Stamp-based protocols, Validation based protocols.

6. Recovery System :

(08)

Failure Classification, The storage Structure, Recovery and Atomicity, Log-Based Recovery, Recovery with concurrent Transactions, Buffer Management, Shadow Paging, Failure with Loss of Non-Volatile Storage.

Text Book :

1. Database System Concept by Henry F. Korth, Abraham Silberschatz, Sudarshan (McGraw Hill Inc.) Fifth Edition
2. Database System Concept by Henry F. Korth, Abraham Silberschatz, Third edition (McGraw Hill Inc.)

Reference Books :

1. Fundamentals of Database Systems – Ramez ElMasri and S B Navathe (Fifth Edition Pearson Education) .
2. Database Management Systems- Ramkrishnan Gehreke (third edition, Tata McGraw Hill).
3. Principles of Database Systems by J.D. Ullman (Galgotia Publications)

Term Work :**Practicals :**

1. It should consist of minimum 8-10 experiments based on above topics and practicals should be implemented in C++/Java as listed below:
 1. Title : ER Diagrams.
Draw ER diagrams (around 5 in number) for any specific application & Convert them into tables.
 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 simulate log based protocol using Immediate Database modification OR Deferred database modification.
 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.
 11. Case Study of DB2 /SQL / Oracle
Implement also Assignment no. 2 to 5 using above mentioned RDBMS package

T.E. (Computer Science & Engineering) Part – II
2. COMPILER CONSTRUCTION

Lectures : 3 hrs/week
Practicals : 2 hrs/week

Theory : 100 marks
T/W : 25 marks

SECTION - I

- 1. Introduction to Compiling:** (3)
Compilers, Phases of a compiler, Compiler construction tools, A simple one pass compiler.
- 2. Lexical Analysis:** (5)
Role of a Lexical analyzer, input buffering, specification and recognition of tokens, finite automata implications, designing a lexical analyzer generator .
- 3. Syntax Analysis:** . (6)
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
- 4. Syntax Directed Translation:** (6)
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.

SECTION - II

- 5. Run Time Environments :** (3)
Source language issues, storage organisation and allocation strategies, parameter passing, symbol table organisations and generations, dynamic storage allocations.
- 6. Intermediate Code Generation :** (3)
Intermediate languages, declarations, assignment statements and boolean expressions, case statements, back patching, procedure calls.
- 7. Code Generation :** (5)
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.
- 8. Code Optimization :** (5)
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.

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. Compiler Construction – Principles & Practice – Ken Louden (Cengage Learning)
3. Compiler Design in C – Allen I. Holub (PHI / Pearson Education)
4. Compiler Construction - Barret, Bates, Couch (Galgotia)
5. Unix Programming - Pepkin Pike.
6. Crafting a compiler with C – Charls Fischer, Richard LeBlane (Pearson Education)

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

Practicals :

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 & Engineering) Part – II
3. OPERATING SYSTEM – II

Lectures : 4 Hrs/Week
Practical : 2 Hrs/ Week

Theory : 100 Marks
Term work : 25 Marks
POE : 50 Marks

SECTION – I

- 1. Introduction :** (7)
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 : Practicals : It should consist of minimum 8-10 experiments based on topics of above syllabus.

T.E. (Computer Science & Engineering) Part – II
4. Software Engineering

Lecture: 4 hrs/week

Theory: 100 Marks

SECTION-I

- 1. Introduction to S/W Engineering. :** (6)
SDLC Definition & Overview-Phase Development process, Software processes, Process, Characteristics of a software process, software development process, project management process, software configuration management process, process management process.
- 2. S/W Requirement Analysis and Specification :** (4)
Software requirements, problem analysis, Requirements specification, validation, Metrics.
- 3. Function Oriented Design** (6)
Design principles, Module level concepts, Design notation and specification, Structured Design Methodology, verification, Metrics.
- 4. The Project Planning** (6)
The project planning infrastructure-Process Database, process capability baseline, process asset and the body of knowledge system.
Effort estimation and scheduling –Estimation and scheduling concepts, effort estimation scheduling
Quality planning-Quality Concepts, Qualitative quality management planning, Defect prevention planning
Risk Management – Concepts of risk and risk management assessment, risk control, the structure of the project management plan.

SECTION-II

- 5. Object Oriented Design** (6)
OO Analysis and OO design, Concepts, Design notation and specification, Design Methodology,UML Diagrams.
- 6. Managing S/W projects** (4)
Processes and project management and the CMM project management process, Training for project managers, SEPG support to projects.
Configuration Measurement- Concept in Config management, The Config management process.
- 7. Project Execution and closure** (4)
Review the Reviews, the review process, Data Collection, Monitoring control
Project Monitoring and control - Project Tracking, milestone analysis. Activity level analysis using SPC, Defect analysis and prevention, Process monitoring and Audit
- 8. Testing** (6)
Black box and White box testing , Object Oriented S/W testing methods, Testing for Quality, Functional testing, Unit testing, System testing, User satisfaction testing, Test cases and Test Plans

Text Books:

1. An Integrated Approach to Software Engineering- 3rd edition: Pankaj Jalote(Narosa Publishers)
2. Software Project management in practice-Pankaj Jalote

References:

1. Software Engineering- Practitioner Approach: Roger S. Pressman. 6th edition.
2. Software Engineering Fundamentals –Ali Behforooz and Frederick j. Hudson (Oxford University Press)

T.E. (Computer Science & Engineering) Part – II
5. ARTIFICIAL INTELLIGENCE

Lectures : 3 hrs/week

Theory : 100 marks

SECTION – I

- 1. Introduction :** (3)
Defination of AI, The AI problems, the understanding Assumption, AI Technique, The level of the model, criteria for success.
- 2. Problem spaces and Search :** (6)
Defination of a problem as a state space search, Production systems, Problem characteristics, Production system characteristics, Issues in the design of search programs
- 3. Heuristic Search Techniques :** (6)
Generate and Test, Hill climbing, Best first search, Problem reduction, Constraint satisfaction, Means – Ends analysis.
- 4. Knowledge Representation :** (3)
Representation and Mappings, Approaches to Knowledge representation, Issues in knowledge representation, The frame problem.

SECTION – II

- 5. Knowledge Representation Techniques :** (5)
Representing simple facts in predicate logic, Representing instances and Isa Relationships, computable functions & Predicates, Resolution, Natural deduction, Procedural versus declarative knowledge, logic programming, Forward versus backward reasoning, matching, control knowledge.
- 6. Symbolic Reasoning under uncertainty and Statistical Reasoning :** (5)
Introduction to non monotonic reasoning, logic for non monotonic reasoning, implementation issues, probability and Baye’s Theorem in certainty factor and Rule – Based systems, Bayesian Networks, Demster Shafer Theory, Fuzzy logic.
- 7. Slot and Filter structures :** (4)
Semantic Nets, Frames, Conceptual dependency, Scripts, CYC.
- 8. Expert System :** (4)
Introduction, Rule Based System Architectures, Non production systems architectures, Dealing with uncertainty , Knowledge Acquisition and validation, Knowledge system building tools.

Text Book :

1. Artificial Intelligence - Elaine Rich, Kevin Knight (Second Edition) (TMGH)
(For Chapter 1 to 7)
2. Introduction to Artificial Intelligence & Expert systems – Dan. W. Patterson (Pearson Education)
(For Chapter 8)

References : -

1. AI : A modern approach - Stuart Russell and Peter Norvig (PHI)
2. Introduction to Artificial Intelligence – Eugene Charniak, Drew Mc Dermott (Pearson Education)

T.E. (Computer Science & Engineering) Part – II
6. Advanced Java

Theory- 2 Hrs/Week
Practical-2 Hrs/Week

POE: 50 Marks
T/W: 25 Marks

- 1. The Java Collection Classes (04)**
Introduction, The Arrays Class, Searching and Sorting Arrays of Primitives, Sorting Arrays of Objects, The Comparable and Comparator Interfaces, Sorting – Using Comparable, Sorting – Using Comparator, Collections, Lists and Sets, Iterators, Lists and Iterators Example, Maps, Maps and Iterators Example, The Collection Class
- 2. Advanced I/O (03)**
Introduction, Basic File I/O Example, Buffered I/O, The Console Class, Object Serialization, Serialization Issues, Writing your own I/O Classes, Property Files
- 3. Remote Method Invocation (RMI) (04)**
Introduction, RMI Architecture, The Remote Interface, The Remote Object, Writing the Server, The RMI Compiler, Writing the Client, Remote Method Arguments and Return Values, Dynamic Loading of Stub Classes, Exercises
- 4. Java Database Connectivity (JDBC) (04)**
Introduction, Relational Databases, Structured Query Language, A Sample Program, Transactions, Meta Data
- 5. Servlets (05)**
Definition, Basic Servlet Application Programming Interface (API), Servlet Architecture, Servlet Lifecycle, Creating Servlet Application file and Deploying application
- 6. JSP (05)**
Introduction to JSP, JSP vs. Servlet, JSP Architecture, Life cycle of JSP, JSP Elements, JSP Documents, Action Elements
- 7. Enterprise Java Beans (05)**
Introduction, Types of Enterprise Bean, Writing Enterprise Beans, Beginning with Enterprise Java Beans, Working with session beans, Working with entity beans
- 8. Struts (03)**
Struts Framework Basic, Understanding Struts, MVC Architecture, Setting up Struts, Struts Flow Control

Text Books:-

1. Java Server Programming for Professionals - Ivan Bayross, Sharanam Shah, Cynthia Bayross and Vaishali Shah, 2nd Edition, Shroff Publishers Distributors Pvt. Ltd.
2. Core Java, Volume II – Advanced Features - Cay S. Horstmann, Gary Cornell, 8th Edition
3. Core Servlets and Java Server Pages, Volume 2, Advance Technology, 2nd Edition – Marty Hall, Larary Brown, Yaakov Chaikin

Reference Books:-

1. Head First Sevlets and JSP – Bryan Bosham, Kathy Sierra, Bert Bates, O’Reily Publication

Termwork : Practicals :

Ten to Fifteen experiments at least one on each topic is expected, where students will be able to use Eclipse, Net Beans and have hands on Strut Programming

- Java SDK 6.0
- Eclipse
- Net Beans Platform
- Tomcat / Web Logic / Websphere

T.E. (Computer Science & Engineering) Part – II

7. SEMINAR

Practicals : 2hrs/week

Term work : 50 marks

The groups of students of strength 4-6 should be formed by the end of T.E.-I. The project areas 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 (whose title is finalized in TE Part – II) 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.