SOLAPUR UNIVERSITY, SOLAPUR

FACULTY OF ENGINEERING & TECHNOLOGY

COMPUTER SCIE[illegible]ND ENGINEERING

Syllabus Structure and detailed syllabus of

T.E. (Computer Science & Engineering)

w.e.f. Academic Year 2016-17

SOLAPUR UNIVERSITY, SOLAPUR
Computer Science and Engineering

Structure of T. E. (Computer Science & Engineering) w. e. f. 2016-17

SOLAPUR UNIVERSITY, SOLAPUR.

Faculty of Engineering & Technology (Revised from 2013-2014)

Credit System structure of T.E. Computer Science and Engineering W.E.F. 2016-2017

Semester I

Theory Course Name	Hrs./week			Credits	Examination Scheme			
	L	T	P		ISE	ESE	ICA	Total
Operating System Concepts	3	1	-	4	30	70	--	100
System Programming	3	-	2	4	30	70	25	125
Computer Networks	4	-	-	4	30	70	--	100
Design and Analysis of Algorithm	3	1	-	4	30	70	25	125
Computer Organization	3	-	-	3	30	70	25	125
Self Learning (HSS)	-	-	-	2	-	50	-	50
Sub Total	16	2	2	21	150	400	75	625
Laboratory/Workshop								
						ESE / POE		
Operating System Concepts	-	-	2	1	-	50	25	75
Computer Networks	-	--	2	1	-	50	25	75
# Lab – Java Programming	2	--	4	4	-	50	25	75
Sub Total	2	-	8	6	-	150	75	225
Grand Total	18	2	10	27	150	550	150	850

Abbreviations: L- Lectures, P –Practical, T- Tutorial, ISE- In Semester Exam., ESE - End Semester Exam, ICA- Internal Continuous Assessment
ISE -Internal Tests, ESE - University Examination (Theory &/ POE &/Oral examination)
Note: 1) '#' indicates Practical exam only.

SOLAPUR UNIVERSITY, SOLAPUR.

Faculty of Engineering & Technology (Revised from 2013-2014)

Credit System structure of T.E. Computer Science and Engineering W.E.F. 2016-2017

Semester II

Theory Course Name	Hrs/week			Credits	Examination Scheme			
	L	T	P		ISE	ESE	ICA	Total
Compiler Construction	3	-	2	4	30	70	25	125
Unix Operating System	3	-	2	4	30	70	25	125
Mobile Computing	3	1	-	4	30	70	25	125
Database Engineering	4	-	-	4	30	70	--	100
Software Engineering	3	1	-	4	30	70	25	125
Self Learning Module-II (HSS/Technical)	--	-	-	-	-	50	-	50
Sub Total	16	2	4	20	150	400	100	650
Laboratory/Workshop								
						ESE		
						POE	OE	
Database Engineering	--	--	2	1	--	50	-	25	75
# Lab - Programming in C#.net	2	--	2	3	--	50	-	25	75
# Mini project	--	--	2	1	--	--	25	25	50
Sub Total	2	--	6	5	--	125		75	200
Grand Total	18	2	10	25	150	525		175	850

Abbreviations: L- Lectures, P –Practical, T- Tutorial, ISE- In Semester Exam., ESE - End Semester Exam, ICA- Internal Continuous Assessment
ISE -Internal Tests, ESE - University Examination (Theory &/ POE &/Oral examination)
Note: 1) '#' indicates Practical exam only.

Subjects for Self Learning for Humanities and Social Sciences (HSS)

1. Economics
2. Psychology
3. Philosophy
4. Sociology
5. Humanities

Subjects for Self Learning for Technical Subjects

1. Computer Modeling and Simulation
2. Software licenses and practices
3. Network set up & management tools

Note:

1. The term-work will be assessed based on continuous internal evaluation including class tests, assignments, performance in laboratories, interaction in class, quizzes and group discussions as applicable.
2. The batch size for 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.
3. Mini Project shall consist of developing small software based on tools & technologies learnt in SE and TE.
4. Project group for T.E.(CSE) Part II Mini Project shall be of 4 / 5 students.
5. Vocational Training (evaluated at B.E. Part-I) of minimum 15 days shall be completed in any vacation after S.E. Part-II but before B.E. Part-I & the report shall be submitted and evaluated in B.E. Part-I.
6. Student shall select one Self Learning Module from HSS at T.E. Part I and oneself learning module either from HSS or from technical at T.E. Part II.
7. Curriculum for Humanities and Social Sciences Self Learning Modules is common for all under graduate programmes under faculty of Engineering and Technology.

Abbreviations: L- Lectures, P –Practical, T- Tutorial, ISE- In Semester Exam., ESE - End Semester Exam, ICA- Internal Continuous Assessment
ISE -Internal Tests, ESE - University Examination (Theory &/ POE &/Oral examination)
Note: 1) '#' indicates Practical exam only.

SOLAPUR UNIVERSITY, SOLAPUR
T.E. (COMPUTER SCIENCE & ENGINEERING)
Semester – I
1. OPERATING SYSTEM CONCEPTS

Teaching Scheme	**Examination Scheme**
Lecture: 3 Hrs/Week	**Theory:** 100 Marks
Practical: 2 Hrs/Week	**Termwork:** 25 marks
Tutorial: 1 Hr/Week	**Practical Oral Examination:** 50 marks

COURSE OBJECTIVES:

1) To expose the importance of the role and structure of operating system.
2) To learn basics of operating system such as Process Management, Memory Management and I/O device management.

COURSE OUTCOMES:

At the end of the course, student will be able to

1) Recognize the role, structure of OS, applications and relationship between them.
2) Analyze the features and functions provided by Operating system modules (such as process control, CPU scheduling, mutual exclusion, deadlock, memory management, synchronization etc.)

SECTION – I

Unit 1: Introduction (5 Hrs.)

Operating system definition, Simple Batch System, Multi programmed Batch System, Time Sharing System, Personal Computer System, Parallel System, Real Time System, and System Calls.

Unit 2: Process (6 Hrs.)

Process Concept, Process Scheduling, Operation on process, Cooperating process, Threads, Interprocess Communication.

Unit 3: Process Scheduling (6 Hrs.)

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

Unit 4: Interprocess synchronization (5 Hrs.)

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

SECTION – II

Unit 5: Deadlocks (7 Hrs.)

System model, Deadlock characterization, Methods for handling deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from deadlock, combined approach to deadlock.

Unit 6: Memory Management (6 Hrs.)

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

Unit 7: Virtual Memory **(5 Hrs.)**

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

Unit 8: I/O system **(4 Hrs.)**

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

Text Books:

1. Operating System concepts – 5th, 7th or 8th Edition – Silberschatz, Galvin (John Wiley).

Reference Books:

1. Operating Systems: Internals and Design Principles, 5th Edition by William Stallings (PHI)
2. Operating system with [illegible] T by Achyut Godbole (TMGH).
3. Operating Systems – De[illegible]tion

Term work:

Tutorials:

In tutorial session, studen[illegible] problems and be guided for the solution

Tutorials are based on:

1. Study and Compar[illegible]
2. Examples of IPC s[illegible]XP
3. Exercise problems [illegible]
4. Exercise problems [illegible]
5. Exercise problems [illegible]
6. Exercise problems [illegible]
7. Disk Performance [illegible]ıg Head Disk Storage, Disk sche[illegible]
8. File System: Intr[illegible]ıization, File Allocation, Free sp[illegible]
9. Exercise problems [illegible]

Practical List:

It should consist of the 12 [illegible]ıx.

1. Study of UNIX Operating System and [illegible]
2. Implementation of a program which describe the use of system calls such as fork (), abort (), suspend () etc.
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.
7. Implementation of Banker's Algorithm for Deadlock Avoidance.
8. Implementation of RAG or WFG method for Deadlock detection for single instance of resources.
9. Simulation of Page Replacement strategies (FIFO, LRU, Optimal) based on Java Multithreading.
10. Implementation of Mutual Exclusion 1st/ 2nd/ 3rd algorithm.
11. Implementation of Mutual Exclusion using semaphore (wait & signal)
12. Implementation of producer consumer problem (Bounded buffer)

SOLAPUR UNIVERSITY, SOLAPUR

T.E. (COMPUTER SCIENCE & ENGINEERING)
Semester – I
2. SYSTEM PROGRAMMING

Teaching Scheme	**Examination Scheme**
Lecture: 3 Hrs/Week	**Theory:** 100 marks
Practical: 2 Hrs/week	**Termwork:** 25 marks

COURSE OBJECTIVES:

1. To learn the principles of processing of an HLL program for execution on a computer system.
2. To design computer language processors.
3. To acquire skills of Language processor development tools.

COURSE OUTCOMES:

At the end of the course, student will be able to

1. Identify various language processors.
2. Design and implement prototypes of language processors.
3. Apply language processor development tools to create Language Processors.

SECTION – I

Unit 1: Language Processors **(8 Hrs.)**

Introduction, language processing activities, Fundamentals of language processing, Fundamentals of language, Specification, language Processor development tools.

Unit 2: Assemblers **(8 Hrs.)**

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.

Unit 3: Macros and Macro Processors **(6 Hrs.)**

Macro definition and call, Macro Expansion, Nested macro calls, Design of Macro preprocessor-Design overview.

SECTION – II

Unit 4: Compilers and Interpreters **(9 Hrs.)**

Aspects of compilation, compilation of expressions, code optimization, Static and dynamic memory allocation, Memory allocation in block structured languages(Scope Rules, Memory allocation and access, Dynamic pointer), Interpreters

Unit 5: Linkers **(6 Hrs.)**

Relocation and linking concepts, design of a linker, Self-relocating programs, linking for overlays.

Unit 6: Loaders **(7 Hrs.)**

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. Dhamdhere (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:

Practical List:

Practical assignments should be carried based on –

1. Simulation of text editor
2. Introduction o
3. Implementatic
4. Implementatic
5. Design and im
6. Design and im
7. Symbol table
8. Design Lex sp umbers, operators, white spaces.
9. Implementatic
10. Simulation of
11. Simulation of

SOLAPUR UNIVERSITY, SOLAPUR

T.E. (COMPUTER SCIENCE & ENGINEERING)

Semester – I

3. COMPUTER NETWORKS

Teaching Scheme
Lecture: 4 Hrs/Week
Practical: 2 Hrs/Week

Examination Scheme
Theory: 100 Marks
Term-Work: 25 Marks
Practical Oral Exam: 50 Marks

COURSE OBJECTIVES:

1) To build the idea of multiple layers in the data communication and the addressing mechanism between the different layers of OSI Reference Model.
2) To introduce the student with client-server paradigm for socket interfaces to discuss the client-server communication using connectionless & connection-oriented services offered by the transport layer protocols.
3) To study the architecture of WWW, HTTP, e-Mail & describe the concepts of hypertext, hypermedia, web clients, web servers and their components to define URL, different Web documents in the application layer.

COURSE OUTCOMES:

At the end of the course, student will be able to

1) To demonstrate the purpose of different layers.
2) To write application layer protocols using services offered by the transport layer protocols such as UDP, TCP & SCTP.
3) To show the functioning of web based mail system and web services working mechanism.

SECTION – I

Unit 1: Overview of TCP/IP Protocol Suite (4 Hrs.)

TCP/IP Protocol Suite: Comparison between OSI & TCP/IP Protocol Suite, Layers in the TCP/IP Protocol Suite, Addressing: Physical, Logical, Port & Application Specific Addresses

Unit 2:Transport Layer (14 Hrs.)

UDP: Overview of the OSI Model and the TCP/IP Protocol Suite, UDP: Introduction, User Datagram, UDP Services, UDP Applications, UDP Package,

TCP: TCP Services, TCP Features, Segment, A TCP Connection, State Transition Diagram, Window in TCP, Flow Control, Error Control, Congestion Control, TCP Timers, TCP Package

SCTP: Stream Control Transmission Protocol: Introduction, SCTP Services, SCTP Features, Packet Format, An SCTP Association, State Transition Diagram

Unit 3: Client Server Model and Socket Interface (8 Hrs.)

Client Server Paradigm: Server, Client, Concurrency, Concurrency in Clients, Concurrency in Servers, Socket, Byte Ordering Functions, Address Transformation Functions, Memory Management Functions, Socket System Calls, Connectionless Iterative Server, UDP Client Server Programs, Connection-oriented Concurrent Server, TCP Client Server Programs.

SECTION – II

Unit 4: Host Configuration & Domain Name System (8 Hrs.)

Host Configuration: BOOTP Operation, Packet format, DHCP: Introduction, DHCP Operation and Configuration.

Domain Name System: Need for DNS, Name Space, DNS In the Internet, Resolution, DNS Messages, Types of Records, Encapsulation, DDNS.

Unit 5: Remote Login and File Transfer (10 Hrs.)

Remote Login: TELNET Concept, Time-Sharing Environment, Network Virtual Terminal, Embedding, Options, Symmetry, Suboption Negotiation, Controlling the Server, Out-of-Band Signaling, Escape Character, Mode of Operation, User Interface.

SSH: Components, Port Forwarding, Format of SSH Packets.

FTP: Connections, Communication, Command Processing, File Transfer, Anonymous FTP, **TFTP:** Messages, Connection, Data Transfer, UDP Ports, TFTP Applications.

Unit 6: WWW, HTTP a (8 Hrs.)

World Wide Web and HT

Electronic Mail: SMTP, P Agent: SMTP, Message Access Agent: PO

Text Books:

1. TCP/IP Protocol Su
2. TCP/IP Protocol Su
3. TCP/IP Protocol Su
4. Computer Network
5. Computer Network ational Edition: James F. Kurose an

Reference Books:

1. Internetworking wit ons: Douglas E. Comer
2. Data and Computer
3. Data Communicatic

Term work:

Student should perform 10 t eferably conducted on Unix / Linux p using C language.

1. Installation of Unix/Linux Operating System.
2. Configuration of Network-Assigning IP Address, Subnet-Mask, Default Gateway, DNS Server Addresses & Testing Basic Connectivity.
3. Study of typical network components such as Server, Client, Network Interface Card, Connector (RJ-45), Communication Medium, Modem, Firewall, Patch Panel, Racks, Leased Line, Connecting Devices: Repeaters, Hubs, Bridges, Switches, Routers.
4. Study of College Network and Design of Any New Network.
5. Connectionless Iterative Server: C Implementation of Client-Server Programs Using Iterative UDP Server.
6. Connection-oriented Iterative Server: C Implementation of Client-Server Programs Using Iterative TCP Server.
7. Connection-oriented Concurrent Server: C Implementation of Client-Server Programs Using Concurrent TCP Server.
8. Implementation of Simple Network Chatting Application.
9. Remote Login: TELNET
    a. Log on to a remote computer from client using TELNET.

b. After logging on executes few commands at remote server from client. For example user wants a server to display a file (hello.txt) on a remote server then he/she types: *cat hello.txt*.
c. Log on to a remote computer from client using TELNET and Putty terminal emulator. After logging on execute few commands. Here Client and Server are on heterogeneous systems, for example client is on windows and server is on Linux.

10. Remote Login: SSH
    a. Log on to a remote computer from client using SSH.
    b. After logging on executes few commands at remote server from client. For example user wants a server to display a file (hello.txt) on a remote server then he/she types: *cat hello*
    c. Log on terminal emulator. After l d Server are on heteroge r is on Linux.
    d. Execute H without logging on. For r one may can use *ssh user*
11. File Transfer: F
    a. Connect
    b. Downlo e (hello.txt) on the server th
    c. Upload lo.txt) on the client then upl
    d. List dire contents of current director
12. Simulation of D
13. Simulation of D
14. Study of the nex

SOLAPUR UNIVERSITY, SOLAPUR

T.E. (Computer Science & Engineering)

Semester - I

4. DESIGN AND ANAYLISIS OF ALGORITHM

Teaching Scheme	**Examination Scheme**
Lecture: 3 Hrs/week	**Theory:** 100 marks
Tutorial: 1 Hr/week	**Term work:** 25 marks

COURSE OBJECTIVES:

1. To study algorithm analysis, design and application.
2. To evaluate and compare algorithms using worst, average and best case analysis.
3. To equip the student with essential algorithm design techniques such as divide and conquer, dynamic programming and the greedy methods and many of its applications.
4. To explain the difference between tractable and intractable problems and identify basic complexity classes such as P, NP complete and NP-hard.

COURSE OUTCOMES:

At the end of the course, student will be able to

1. Analyze the asymptotic performance of algorithms.
2. Demonstrate a familiarity with major algorithms.
3. Apply important algorithmic design paradigms and methods of analysis.
4. Synthesize efficient algorithms in common engineering design situations.

SECTION – I

Unit 1: Introduction **(8 Hrs.)**

Algorithm Specification: Pseudo code Conventions, Recursive Algorithm
Performance Analysis: Space Complexity, Time Complexity, Calculating worst case and average case complexities: Case study, Amortized Complexity, Asymptotic Notations, Practical Complexities, Performance Measurement

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

The general method, Binary search, Finding the maximum and minimum, Merge sort, Quicksort, Selection, Strassens Matrix multiplications

Unit 3: The Greedy method **(8 Hrs.)**

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

Unit 4: Dynamic Programming **(8 Hrs.)**

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

Unit 5: Backtracking **(7 Hrs.)**

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

Unit 6: NP-Hard and NP-Complete problems **(7 Hrs.)**

Tractable and Intractable Problems: Computability. The Halting problem, Computability classes – P, NP-complete and NP-hard, Cook's theorem, Standard NP-complete problems Reduction techniques

Text Books:

1. Fundamentals of Computer Algorithms–Horowitz, Sahni & Rajasekaran (Galgotia Publications)
2. Algorithm Design –Michael T. Goodrich, Roberto Tamassla, Wiley Student Edition
3. Data Structu ison wesley)

Reference books:

1. Fundamental o Publication)
2. Introduction to
3. Introduction to IcGrawhill)
4. Algorithm Des

Termwork:

Assignment List:

1. Assignment on S
2. Problems on Asy
3. Performance mea
4. Assignment on R
5. Study and solve p
6. Finding time com
7. Solve exercise on
8. Assignment on Dynamic programming
9. Assignment on Backtracking
10. Exercise on NP-Hard and NP-Complete problems

SOLAPUR UNIVERSITY, SOLAPUR

T.E. (Computer Science & Engineering)

Semester - I

5. COMPUTER ORGANIZATION

Teaching Scheme	**Examination Scheme**
Lecture: - 3 Hrs/Week	**Theory –** 100 Marks
	Term-Work – 25 Marks

COURSE OBJECTIVES

1) To learn fundamentals of computer organization.
2) To know the processor level design, memory and I/O organization.
3) To acquire fundamentals of pipelined architecture.

COURSE OUTCOMES

At the end of the course, student will be able to

1) Justify the principles of computer organization.
2) Identify performance of processor, design memory hierarchy and interface I/O devices.
3) Identify parallel architecture.

Course Instruction

Concepts of CPU may be clarified through simulation software tools.

SECTION I

Unit 1: **Basic Structure of Computer Hardware** **(06 Hrs)**

Functional Units, Basic operational concepts, Bus Structure, Generation of computers.

Unit 2: Processor Level Design **(11 Hrs.)**

Instruction format, Instruction types, Bus hierarchical architecture, RISC, CISC, Fixed point arithmetic-Addition, Subtraction, Multiplication (Booth Algorithm),Fast multiplication, Division(Restoring and Non Restoring Algorithm) Implementation of floating point operation, IEEE floating point standard

Unit 3: Hardwired Control Unit **(05 Hrs.)**

Hardwired Control Unit: Design Methods (Sequence counter); Multiplier Control Unit (Introduction), (Implementation of Multiplier in each case**).**

SECTION II

Unit 4: Memory Organization and Design **(08 Hrs.)**

Virtual memory: Memory Hierarchy, Main memory allocation, Segments & pages, Replacement policies ,High Speed memories-Interleaved memories, Cache, Associative**.**

Unit 5: Input-Output Organization **(05 Hrs.)**

Accessing I/O devices, Direct Memory Access, Interrupt Handling, I/O Interfaces, I/O Channels

Unit 6: Parallel Processing and Pipelined Architecture **(09 Hrs.)**

Uniprocessor and Multiprocessor parallelism; Types of uniprocessor parallelism; Basics of Pipelining & vector processing, Multiprocessor Architecture-tightly coupled & loosely coupled , Linear and Nonlinear pipeline ,Pipeline hazards.

Text Books:

1. Computer Architecture & Organization – J.P.Hayes (MGH) (Chapters:1,2,3,4)
2. Computer Organization - Hamacher and Zaky (MGH)(Chapters:1,5)
3. Advanced Computer Architecture and Parallel Processing- Kai Hwang and Briggs (MGH) (Chapter:6)
4. Computer System Architecture and Organization-Dr.M.Usha and T.S.Srikanth (WILEY INDIA)

Reference Books:

1. Advanced Computer Architecture- Kai Hwang (MGH)
2. Computer Organization and Architecture –Hennessy Patterson (ELSEVIER)
3. Design for Performance-William Stallings (PEARSON)

Termwork:

Assignment List:

1. Discuss the different tions of computer and compa
2. Name the different a unctions with an example.
3. Explain IEEE standa single precision and double
4. Hexadecimal values in single precision format.
5. With an example dis oring method.
6. Why program contro ne reasons.
7. Discuss the sequence om a single device.
8. What are the major c condary memory hierarchies?
9. With an example, ex vo different memory capacities.
10. Draw the architectur .
11. What is hazard? Wh example. What are types of data hazard? With example, explain types of data hazard.

SOLAPUR UNIVERSITY, SOLAPUR

T.E. (Computer Science & Engineering)

Semester - I

6. Lab - JAVA Programming

Teaching Scheme	**Examination Scheme**
Lecture: 2 Hrs/Week	**Term-Work:** 25 Marks
Practical: 4 Hrs/Week	**Practical Oral Exam**: 50 marks

COURSE OBJECTIVES:

1 To learn Object oriented programming paradigms using Java language.
2 To introduce the student Basic Java API Classes and Features for use in Application programming.
3 To impart basic understanding and analyze platform independent application runtime environment to create standalone GUI, Web applications using Java language.

COURSE OUTCOMES:

At the end of the course, student will be able to

1 Implement Object oriented programming paradigms using Java language.
2 Explore and use the Java APIs for implementing various functionalities of an Application.
3 Analyze platform independent application runtime environment and choose appropriate runtime environment to create GUI and Web applications using Java language.

SECTION I

Unit 1: Basics and Fundamentals of Java (4 hrs)

Basics: Java Runtime Environment. Languages Basics: Variables, Operators, Expressions, Statements, Blocks, Control flow Statements.

Fundamentals: Data Types, Arrays, Objects and Classes. Fields and Methods, Access control, Modifiers, Constructors, Overloading methods, Abstract classes, Nested classes, Packages, Wrapper classes, Interfaces, Using the Keyword "this" . Object Life time & Garbage Collection. Recursion in Java.

Unit 2: Inheritance, Numbers and Strings, Generics (4 hrs)

Inheritance: Extending Classes and Inheritance, Types of Inheritance in Java, Polymorphism, Type Compatibility and Conversion, Overriding and Hiding Methods, Hiding Fields, Using the Keyword "super"

Numbers and Strings: String Class and Methods, StringBuffer Class and Methods

Generics: Generic Classes and Methods.

Unit 3: Exceptions, Error Handling and Basic IO (5 hrs)

Exceptions and Error Handling: Exceptions and Errors, Catching and Handling Exceptions The try Block, The catch Blocks, The finally Block, Specifying the Exceptions Thrown by a Method, Throwing Exceptions, Chained Exceptions , Creating Exception Classes, Checked and Unchecked Exceptions, Advantages of Exceptions.

Basic I/O: I/O Streams, Byte Streams, Character Streams, Buffered Streams, Scanning and Formatting, Data Streams, Object Streams , File I/O Classes: Reading, Writing, and Creating

Files and Directories.

Unit 4: Java Collections Framework **(3 hrs)**
Introduction, The Arrays Class, Searching and Sorting arrays of primitive data types, Sorting Arrays of Objects, The Comparable and Comparator Interfaces, Sorting using Comparable & Comparator, Collections: Lists, Sets, Maps, Trees, Iterators and Collections, The Collection Class.

SECTION II

Unit 5: Multithreading and Network Programming **(4 hrs)**
Multithreading: Creating Threads, Thread scheduling and priority, Thread interruptions and synchronization, Thread Safety, Pros and Cons of Multithreading.
Network Programming: Networking fundamentals, TCP, UDP communication in Java. Client server programming: InetAddress, URLs, Sockets, DatagramSockets.

Unit 6: JDBC and RMI **(4 hrs)**
JDBC: Introduction to JD Using JDBC API.
RMI: Introduction, RMI bject, Writing the Server and Client, Re and Skeleton Classes.

Unit 7: GUI Programmi **(4 hrs)**
GUI Programming with T and Swing package, , Layouts, Event nponents.
Applets: Introduction, veloping and Deploying Applets

Unit 8: Servlets and JSP **(4 hrs)**
Introduction to Servlets a hitecture and lifecycle. JSP Elements. R , Cookies and Session Handling using Se

Text Books:
1. Head First Java – Kathy
2. The Java™ Programmi olmes, Pearson Publication
3. Head First Servlets and JSP – Bryan Bosham, Kathy Sierra, Bert Bates, O'Reily Publication
4. Core Java for Beginners- Rashmi Kanta Das, Vikas Publishing House Pvt Ltd.

Reference Books:
1. The Java Language Specification, Java SE 7 Edition Book by James Gosling, Oracle Inc. (e-Resource: http://docs.oracle.com/javase/specs/)
2. Java: The Complete Reference 8 Edition - Herbert Schildt , Tata McGraw - Hill Education
3. The Java™ Tutorials. Oracle Inc. (e-Resource: http://docs.oracle.com/javase/tutorial/)
4. Java Server Programming for Professionals - Ivan Bayross, Sharanam Shah, Cynthia Bayross and Vaishali Shah, Shroff Publishers and Distributors Pvt. Ltd, 2nd Edition

Term Work:

- Students should undertake minimum 20 practical assignments based on each above topic.
- The assignments should test and develop student's practical proficiency and ability to use Java API Classes efficiently in writing effective code for varied applications scenarios & requirements.
- Use of IDEs like BlueJ, Eclipse, Netbeans for Interactive development and debugging of Java applications is highly recommended to enhance hands on skills in Java Programming of Students.
- Preferably use Apache Tomcat/GlassFish Server with Eclipse or Netbeans for assignments based on Servlets and JSP.

SOLAPUR UNIVERSITY, SOLAPUR
T.E. (Computer Science & Engineering)
Semester-I
7. Self Learning (HSS)

Examination Scheme
Theory: 50 Marks

Refer to the syllabus common to all programmes under faculty of engineering.

SOLAPUR UNIVERSITY, SOLAPUR

T.E. (Computer Science & Engineering)

Semester-II

1. COMPILER CONSTRUCTION

Teaching Scheme	**Examination Scheme**
Lecture: 3 Hrs/Week	**Theory -** 100 Marks
Practical: 2 Hrs/Week	**Term-Work –** 25 Marks

COURSE OBJECTIVES:

1. To introduce principal structure of compiler, basic theories and methods used for different parts of compiler.
2. To impart knowledge of fundamentals of language translator, structure of a typical compiler, parsing methods etc.
3. To design various phases of compiler such as Lexical analyzer, parser etc.
4. To distinguish different optimization techniques in the design of compiler.

COURSE OUTCOMES:

At the end of the course, student will be able to

1. Apply techniques for the structure of compiler.
2. Use simulation software to justify compiler design.
3. Implement various phases of compiler.
4. Apply different optimization techniques in the design of compiler.
5. Analyze and compare various compilers to select optimum.

Course instructions:

Study of open source software – **YASS** by University of Wisconsin-Madison, to better understand each phase of compiler.

SECTION - I

Unit 1: Introduction: (3 Hrs.)

Language Processor, Structure of Compiler

Unit 2: Lexical Analysis: (5 Hrs.)

The Role of the Lexical analyzer, Input Buffering, Specification of Tokens, Recognition of Tokens, Finite Automata, Conversion of NFA to DFA, Designing a Lexical Analyzer Generator- The Structure of the Generated Analyzer, Pattern Matching based on NFA's, DFA for Lexical Analyzer.

Unit 3: Syntax Analysis: (8 Hrs.)

Introduction, Context Free Grammars, Writing a Grammar, Top-down Parsing, Bottom-Up Parsing, Introduction to LR parsing: Simple LR, More powerful LR parsers

Unit 4: Syntax Directed Translation: (6 Hrs.)

Syntax Directed Definitions, Evaluation Order for SDD's, Application of Syntax Directed Translation, Syntax Directed Translation Scheme, Bottom-up Parsing of L-Attributed SDD's.

SECTION - II

Unit 5: Intermediate Code Generation: (6 Hrs.)

Variants of Syntax Trees, Three Address Code, Types & Declaration, Control Flow – Boolean Expression, Short Circuit Codes & Flow of Control Statements, Backpatching, Switch-Statements.

Unit 6: Run Time Environments: (3 Hrs.)

Storage Organization, Stack Allocation of Space.

Unit 7: Code Generation: (7 Hrs.)

Issues in the Design of a Code Generator, The Target Language, Basic Blocks & Flow Graphs, Simple Code Generator. Register Allocation & Assignment.

Unit 8: Code Optimi (7 Hrs.)

The Principle Sou Blocks, Peephole Optimization, Introdu

Text Book:

1. Compilers - Princi ho, Monica S. Lam, R. Sethi and J.D
2. Compiler Construc

References:

1. Compiler Construc Learning)
2. Compiler Design in
3. Compiler Construc
4. Crafting a compile Education)

Term Work:

It should consist of m Following experiments may be c

Practical List

1. Design a lexic
2. Implement a recognizer for the language in 1.
3. Recursive Descent Parser
4. Shift Reduce Parser
5. Operator Precedence Parser
6. Generate a symbol table for the language given in 1.
7. Generate 3 address codes for the language given in 1.
8. Implement code optimization techniques on the code produced in 7.
9. Generate target code for the code optimized in 4, considering the target machines to be X86.
10. Code Optimization Tools

SOLAPUR UNIVERSITY, SOLAPUR

T. E. (Computer Science & Engineering)

Semester – II

2. UNIX OPERATING SYSTEM

Teaching Scheme	**Examination Scheme**
Lecture: 3 Hrs/Week	**Theory :** 100 Marks
Practical: 2 Hrs/ Week	**Term work:** 25 Marks

COURSE OBJECTIVES:

Teacher needs to focus on and make students learn about some of the following things but not limited to:

1. To introduce fundamentals and architecture of UNIX Operating system including file management, process management, memory management and I/O subsystem of UNIX.
2. To provide hands on commands of UNIX and Shell Programming
3. To build the concept of multiuser operating system.

COURSE OUTCOMES:

At the end of the course, student will be able to

1. Illustrate File Structure, Process Management and Memory Management of Unix using UNIX Architecture
2. Apply basic UNIX/Linux commands , system calls and SHELL Programming
3. To compare between single user and multiuser system

SECTION – I

Unit 1: Introduction (6 Hrs.)

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.

Unit 2: The Buffer Cache (6 Hrs.)

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 (4 Hrs.)

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.

Unit 4: System calls for the file System (5 Hrs.)

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

Unit 5: The Structure of process **(4 Hrs.)**

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 6: Process Control **(6 Hrs.)**

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.

Unit 7: Process Scheduling and Time **(3 Hrs.)**

Process Scheduling, system call for time, clock.

Unit 8: Memory management policies **(5 Hrs.)**

Swapping, Demand p pping

Unit 9: The I/O Sub **(4 Hrs.)**

Driver interfaces, dis

Text Books:

1. The design o
2. Unix Manual

Reference books:

1. Unix concept MGH).
2. Advanced Pr evens.
3. UNIX Conce

Term Work:

It should consist o topics. Following experiments may be

Practical List:

1. Study of Uni
2. Write a progr
3. Write a progr
4. Write a program to implement ls command
5. Write a program to implement ***getblk*** algorithm
6. Write a program to implement ***ialloc & ifree*** algorithm.
7. Write a Program to implement alloc and free algorithm.
8. Study of System calls STAT & FSTAT , PIPES,LINK &UNLINK,DUP,MOUNT &UNMOUNT.
9. Study of shell programming
    - WAP to find whether entered number is even or odd
    - WAP to find factorial of number
    - WAP to find whether entered number is prime or not
    - WAP for fibonnaci series
    - WAP to find sum of series of entered number
    - WAP to find power of number.
10. WAP to implement ***malloc*** algorithm .

SOLAPUR UNIVERSITY, SOLAPUR

T.E. (Computer Science and Engineering)

Semester-II

3. MOBILE COMPUTING

Teaching Scheme	**Examination Scheme**
Lecture: 3 Hrs/Week	**Theory:** 100 Marks
Tutorial: 1 Hr/Week	**Term-Work:** 25 Marks

COURSE OBJECTIVES:

1. To introduce concepts and principles of mobile computing.
2. To explore skills of finding solutions for mobile computing applications.
3. To get acquainted with basics of Android Operating System and its architecture.
4. To introduce NFC standards and practices.

COURSE OUTCOMES:

At the end of the course, student will be able to

1. Apply the principles of mobile computing in the real time.
2. Analyze requirements of mobile compatible applications.
3. Put the basic knowledge gained, into practice in developing mobile based applications using Android.
4. Analyze various scenarios and environments, where NFC can be put into practice.

Course Instruction

Visit to BSNL for practical working of wired and wireless communication system.

SECTION I

Unit 1: **Introduction to Mobile Communication** **(3 Hrs.)**

Analog Communication: Carrier signal, AM, FM, PM, Demodulation, Generations: 1G, 2G, 3G and 4G

Unit 2: Wireless Transmission **(6 Hrs.)**

Frequencies for radio transmission, Signals, Antennas, Signal propagation, Multiplexing, Modulation, Spread spectrum, Cellular system, SDMA, FDMA, TDMA,CDMA.

Unit 3: GSM **(7 Hrs.)**

Mobile services, System architecture, Radio interface, Protocols, Localization and calling, Handover, Security, New data services, Mobile Number Portability.

Unit 4: Wireless LAN **(7 Hrs.)**

IEEE 802.11, Personal Area Network, IEEE 802.15.1 and IEEE 802.15.4 (Bluetooth and ZigBee), Ad-hoc and Sensor network-Introduction, Characteristics of MANET and Applications.

SECTION II

Unit 5: Mobile Network Layer **(7 Hrs.)**

Mobile IP, DHCP

Unit 6: Mobile Transport Layer **(7 Hrs.)**

Traditional TCP, Indirect TCP, Snooping TCP, Mobile TCP, Fast and selective retransmission and recovery, Transaction oriented TCP.

Unit 7: Android OS (Case Study) (5 Hrs.)
Introduction, History, Features and Characteristics, Ecosystem, Hardware Requirements, Development Model, Android Concepts, Overall Architecture.

Unit 8: Near Field Communication (Case Study) (3 Hrs.)
Towards NFC Era, Ubiquitous Computing , Technological Motivation of NFC, RFID and NFC, General architecture of NFC enabled mobile phones.

Text Books:
1. Mobile Communications – Jochen Schiller (PEARSON) (Chapters: 2,3,4,5,6)
2.Introduction to Wireless and Mobile System-D.P.Agrawal and Qing-AnZeng (CENGAGE) (Chapter:1,4)
3.Embedded Android-Porting, Extending, and Customizing- Karim Yaghmour (O'Reilly Media) (Chapter:7)
4: Near Field Communication : From Theory to Practice Vedat Coskun, Kerem Ok, Busra Ozdenizci. (Wiley) (

Reference Books:
1. Wireless Communi ort (PEARSON)
2. Mobile and Persona –(PHI)
3. Mobile Computing oke K Talukder, Hasan Ahmed and I

Termwork:
Assignment List:
1. The message signa Cos(2πfct).Using amplitude modulatio
2. Compare and discus chniques.
3. A TDMA system us frame.
   a) What is the raw da
   b) If guard time and ic efficiency.
4. Give reasons for a it. What are the typical steps for han
5. Which resources ne on using HSCSD or GPRS respectivel
6. How do IEEE 802 hidden terminal problems?
7. List the entities of mobile IP and describe data transfer from a mobile node to a fixed node and vice versa. Why and where is encapsulation needed?
8. What is the basic purpose of DHCP? Name the entities of DHCP. How can DHCP be used for mobility and support of mobile IP?
9. How and why does I-TCP (Indirect TCP) isolate problems on the wireless link? What are the main drawbacks of this solution?
10. Write a case study for selected Android OS of specific version.
11.Case study of NFC in modern smart phone mobiles.

SOLAPUR UNIVERSITY, SOLAPUR
T.E. (Computer Science & Engineering)
Semester-II
4. DATABASE ENGINEERING

Teaching Scheme	**Examination Scheme**
Lecture: 4 Hrs/Week	**Theory:** 100 Marks
Practical: 2 Hrs/Week	**Termwork:** 25 Marks
	Practical Oral Exam. 50 Marks

COURSE OBJECTIVES:

1. To develop the relational model of data,
2. To introduce the students an overview of the database-design process, with E-R model and develop query writing skills in SQL.
3. To familiarize the students with concept of normalization of database.
4. To express the fundamentals of a transaction-processing system and concurrency control.

COURSE OUTCOMES:

At the end of this course, the student will be able to,

1. Apply the basic concepts of relational data model, entity-relationship model, relational database design, relational algebra and database language SQL.
2. Design E-R diagrams to represent simple database for any real time application and formulate SQL queries on it.
3. Design a database, analyze it and improve the design by normalization.
4. Demonstrate knowledge of ACID properties of a transaction and several techniques of concurrency control.

SECTION-I

Unit 1: Introduction (4 Hrs)

Database-System Applications, Purpose of Database Systems, View of Data, Database Languages, Relational Databases, Database Design, Data Storage and Querying, Transaction Management, Database Architecture, Data Mining and Information Retrieval, Specialty Databases, Database Users and Administrators, History of Database Systems.

Unit 2: Relational Model (10 Hrs)

Structure of Relational Databases, Database Schema, Keys, Schema Diagrams, Relational Query Languages, Relational Algebra, Tuple Relational Calculus, Domain Relational Calculus.

Structured Query language (SQL)-Overview, SQL Data Definition, Basic Structure of SQL Queries, Additional Basic Operations, Set Operations, Null Values, Aggregate Functions, Nested Subqueries, Modification of the Database, Join Expressions, Views, Transactions, Integrity Constraints, SQL Data Types and Schemas, Authorization.

Unit 3: Database Design and the E-R Model (6 Hrs)

Overview of Design Process, The Entity-Relationship Model, Constraints, Removing Redundant Attributes In Entity Sets, E-R Diagrams, Reduction to Relational Schemas, E-R Design Issues, Extended E-R Features.

Unit 4: Relational Database design **(8 Hrs)**

Features of Good Relational Designs, Atomic Domains and First Normal Form, Decomposition Using Functional Dependencies, Functional Dependency Theory, Algorithms for Decomposition, Decomposition using Multivalued Dependencies.

SECTION - II

Unit 5: Indexing and Hashing **(7 Hrs)**

Basic Concepts, Ordered Indices, B^+-Tree Index Files, B+-Tree Extensions, B Tree Index Files, Multiple Key Access, Static Hashing, Dynamic Hashing, Comparison of Ordered Indexing and Hashing, Bitmap Indices, Index Definition in SQL.

Unit 6: Transactions **(7 Hrs)**

Transaction Concept, A Sir ın Atomicity and Durability, Transactio l Atomicity, Transaction Isolation Lev ons as SQL Statements.

Unit 7: Concurrency Con **(6 Hrs)**

Lock-Based Protocols, I stamp-Based Protocols, Validation-Base

Unit 8: Recovery System **(8 Hrs)**

Failure Classification, St thm, Buffer Management, Log-Based Concurrent Transactions, Buffer Manag

Text Books:

1.Database system concepts ıan (McGraw Hill Internatio

2. Database system concep shan (McGraw Hill Internatio

3. Database system conc rning) ninth edition.

Reference books:

1. Fundamentals of Database systems by Ramez ElMasri, S. B. Navathe (Pearson Education) fifth edition.
2. Database Management Systems by Ramkrishnan Gehreke (Tata McGraw Hill) third edition.
3. Principles of Database Systems by J. D. Ullman (Galgotia Publications)
4. Advanced Database Management System by Rini Chakrabarti, Shilbhadra Dasgupta (Dreamtech Press Publication).

Course Instructions:

Assignments 1 to 6 should be implemented in PostGreSQL/MySQL/Oracle.
Assignments 7 to 11 should be implemented in C++/Java.

Term Work:

It should consist of 8-10 laboratory assignments as follows:

1. E-R Diagrams (around 5 in number) for any specific application and create a data dictionary for the same.
2. Basic SQL-write simple queries in SQL on the schema created for a specific application.
3. a) More SQL: Aggregates-write queries in SQL using aggregates, grouping and ordering.
   b) Nested sub queries and SQL updates: write queries in SQL using concept of nested sub queries and SQL update commands.
4. a) SQL DDL s.
   b) Schema schema to create const
5. Convert the
6. Write a Jav nts) connectivity using JDBC
7. Write a pr on the database previously c
8. Write a pr base previously created.
9. Write a pr iate or deferred database mc
10. Write a prog .
11. Given a se and closure of functional d

SOLAPUR UNIVERSITY, SOLAPUR

T.E. (Computer Science and Engineering)

Semester - II

5. SOFTWARE ENGINEERING

Teaching Scheme	**Examination Scheme**
Lecture: 3 Hrs/week	**Theory:** 100 Marks
Tutorial: 1 Hr/week	**Term Work:** 25 Marks

COURSE OBJECTIVES

The Course should enable the student

1. To focus on the study of plan, design, architecture and modeling structure layout of software.
2. To illustrate and compare use of life cycle models of software development.
3. To enable the students to analyze and estimate the cost, effort of software product.
4. To learn to embed various quality standards in the software.

COURSE OUTCOMES

At the end of the course, student will be able to

1 Develop the software project using appropriate phases.
2 To implement life cycle models in software development and for their projects.
3 To enhance the quality of product and should be able to apply testing of software.
4 Know the basics of software metrics and result assessment and basics of process improvement.

SECTION-I

Unit 1: Introduction to Software Engineering: **(6 Hrs.)**

SDLC Definition, Phased Development Process, Software processes, Characteristics of a software process, Software Development Process Models.

Overview: Project Management Process, Software Configuration Management process, Process Management Process.

Unit 2: Software Requirements Analysis and Specification: (4 Hrs.)

Software Requirements, Problem Analysis, Requirements Specification, Validation, Metrics.

Unit 3: Function and Object Oriented Design (8 Hrs.)

Design Principles, Module Level Concepts, Design Notation and Specification, Structured Design Methodology, Verification, Metrics.

OO Analysis and OO design Concepts, Design Notation and Specification, Design Methodology, UML Diagrams.

Unit 4: The Project Planning (5 Hrs.)

The Project Planning Infrastructure-Process Database, Process Capability Baseline, Process, Asset and the body of knowledge system, Requirement Change management, Effort Estimation and Scheduling –Concepts, Effort estimation, Scheduling.

SECTION-II

Unit 5: Quality planning, Risk Management and Tracking (5 Hrs.)

Quality Concepts, Qualitative Quality Management Planning, Defect Prevention Planning, Concepts of Risk and Risk Management Assessment, Risk Control, Concepts in Measurements: Measurements, Project Tracking

Unit 6: Agile Project Management (5 Hrs.)

Introduction to APM, Implementation, Iterative Project Management Life Cycle, Adaptive Project Management Life Cycle, Adaptive & Integrating the APM toolkit

Unit 7: Managing Software projects, Project execution and closure (8 Hrs.)

Processes and Project Management, Project Management and The CMM, Team Management, Customer Communication and Issue Resolution, The structure of the Project Management Plan, ation Management Process, Reviews P revention, Project Closure Analysis.

Unit 8: Testing (5 Hrs.)

Testing Fundamental

Text Books:

1. An Integrated A kaj Jalote (Narosa Publishers)
2. Effective Project Wysocki WILEY INDIA, 6th editio
3. Software Project I
4. Software Enginee an, R. S. Pressman & Associates, Inc

Reference Books:

1) Ian Sommerville, ion
2) Software Engine k j. Hudson (Oxford Universi
3) PANKAJ JALOT y Precise Textbook, WILEY INDIA)
4) Software Engineering by Ian Sommerville.

Termwork :

Tutorial List:

Implémentation of mini software projets by applying SDLC cycles.
It should consist of minimum 6 - 8 assignments based on each topic of above syllabus.

SOLAPUR UNIVERSITY, SOLAPUR

T.E. (Computer Science and Engineering)

Semester - II

6. Lab. – Programming in C#.Net

Teaching Scheme	**Examination Scheme**
Lecture: 2 Hrs/week	**Termwork:** 25 Marks
Practical: 2 Hrs/week	**Practical/Oral Exam:** 50 Marks

COURSE OBJECTIVES:

1. To introduce .NET Programming using the C# programming language.
2. To develop basic understanding of the syntactical features of C# programming language and effective use of .NET runtime library APIs to develop robust software applications.
3. To develop ability to design and build Object Oriented and GUI, Web applications on Windows platform.

COURSE OUTCOMES:

At the end of the course, student will be able to

1. Use .NET Framework in building robust software applications using C# programming language.
2. Design and develop Object Oriented and GUI, Web application on Windows platform.

SECTION I

Unit 1: Introduction to .NET Framework (3 Hrs)

The .NET architecture , The common language runtime (CLR) , the, Microsoft intermediate Language code (MSIL),Just in time Compliers , The framework class library, ,The common language specification, common language type system (CTS), Introduction to Visual Studio .NET and Sharp Develop IDE.

Unit 2: C# Application Basics and Language fundamentals (4 Hrs)

Creating and compiling C# programs using command line compiler (csc.exe), Creating applications using IDEs, Namespaces , the "using" keyword , Basic data types, Operators, Flow control and conditional statements , loops, Arrays ,Classes and Objects, Constructor overloading, Methods, Fields, Properties, Access Modifiers and Accessibility Levels, Static methods and fields, Garbage Collection, Structures , Nested Classes, String Manipulations, Naming Conventions, Java vs. C#

Unit 3: Object Oriented Programming using C# (4 Hrs)

Objects and Reference Types, Inheritance, Interfaces and Abstract Classes, Polymorphism, the "virtual" and "override" keyword, the "base " keyword, the "sealed " keyword, The Object Class, the "new" keyword in context of method overriding , Type Casting: Up casting and Down casting, the "is" and "as" keywords, Boxing and Unboxing,

Unit 4: Exception Handling, Events and Delegates (4 Hrs)

Need for Exceptions, Exception Hierarchy, Handling Exceptions using try-catch-finally blocks, creating and defining Custom Exceptions, the "throw" keyword. Events and Delegates in C#, Multicast Delegate, Event Handling

SECTION II

Unit 5: Multithreading and Basic IO in C# **(4 hrs)**

What is Multithreading , Multithreading in C#, Static and Instances members of Thread Class, Basic Thread operations, Thread priorities, Thread Synchronization, **File System and Streams:** Streams and System.IO namespace , Console IO ,Reading writing and updating files and directories, System.IO.FileInfo Class , Serialization and Deserialization.

Unit 6: GUI Programming in C# **(4 hrs)**

Windows Forms and System Windows, Form namespace, Building Windows Forms Applications using IDE, Windows Form controls , Event Handling , List Box , Combo Box, Tree View, File Dialog, Tool Bar, Windows standard Dialog Boxes, Menu Bar, GDI+ Graphics: Drawing Lines ,shapes and images.

UNIT 7: Data ac **(4 hrs)**

Introduction to AI ble, Data Row, Data Column and other ; ADO.NET**.**

UNIT 8: Introdu **(4 hrs)**

Introduction to AS Web Forms, Server Controls, Web app g ASP.NET and C#.

Textbooks:

1 Professional C Karli Watson, Jay Glynn, Morgan

2 Programming i iraw - Hill Education

Reference Books:

1 C# Language S (E-Resource av

2 C# Programmi m/en-US/)

3 Microsoft Visu .

Term Work:

- Students sh based on each above topic.
- The assignments should test and develop student's practical proficiency and ability to use .NET framework libraries and APIs efficiently in writing C# code for varied applications scenarios & requirements.
- Use of IDEs like SharpDevelop and Visual Studio Express Edition for Interactive development and debugging of C#.NET applications is highly recommended to enhance hands on skills in C#.NET Programming of Students.

SOLAPUR UNIVERSITY, SOLAPUR

T.E. (Computer Science and Engineering)

Semester - II

7. Mini Project

Teaching Scheme
Practical – 2 Hrs/week

Examination Scheme
Termwork: 25 Marks
Oral Exam: 25 Marks

COURSE OBJECTIVES:

1. To undertake investigation of complex problems.
2. To motivate students to undertake design of a product, which is sustainable and meaningful to society
3. To enable students to acquire and develop professional skills.
4. To make students learn to work in team.
5. To encourage independent critical thinking, creativity and discipline.
6. To use modern tools and simulation packages
7. To prepare students to implement their acquired engineering knowledge for society.

COURSE OUTCOMES:

At the end of the course, student will be able to

1. Identify and define the problem.
2. Develop a sustainable product or offer a effective solution to industrial problem.
3. Present proposal within budgetary and time constraints with effective communication and writing skills.
4. Develop leadership qualities.
5. Criticize and refine own solution or product.
6. Apply modern tools and simulation packages to develop product.
7. Develop a strong sense of social responsibility and accountability.

Note:

1. There should be a group of preferably 4 students.
2. Students should be given projects in hardware, software, embedded or any contemporary topic in CSE and/or IT
3. One guide should be allocated per batch.

Mini Project ideas (but not limited to):

1. Online Examination module (Multiple choice questions)
2. Attendance recording and analysis software module
3. Examination Result analysis software module
4. Hardware exhibitors such as display board exhibiting all types of mouse / keyboards, HDDs, Monitors etc.), Internal architecture and working
5. Departments / College website
6. Library Management System
7. Hotel Management System
8. Time table generation
9. CD Library management system
10. Admission procedure automation

11. Online passport registration automation
12. Student Feedback system automation
13. Ice Cream parlor management system
14. Pizza hut – account management system
15. Multi player strategy game – Project ideas on Visual Basic, Java, Database
16. A speech response application using some hardware interface using the Microsoft SAPI SDK
17. LAN administrator tool (socket programming comes easy in VB) which will monitor application on a LAN and provide functions.
18. Voice mail systems
19. Computer telephony integration
20. Student Informat
21. Traffic Control s
22. Airline reservatic
23. Simulation for B
24. Mini Calculator i
25. Moving ball gam
26. Tic-tac-toe game
27. Design a persona

SOLAPUR UNIVERSITY, SOLAPUR

T.E. (Computer Science and Engineering)

Semester - II

1. Self Learning (Technical) – 1. Computer Modeling and Simulation

Examination Scheme

Theory: 50 marks

COURSE OUTCOMES:

At the end of the course, student will be able to

1. Learn Modeling and Simulation.
2. Use API libraries for Network Simulator.
3. Perform a task completely on Network Simulator.

Unit 1: **Introduction to Modeling and Simulation :**

When simulation is the appropriate tool and when it is not appropriate; Advantages and disadvantages of Simulation; Areas of application; Systems and system environment; Components of a system; Discrete and continuous systems; Why and what to Model, Model of a system, Types of Models,

Unit 2: Case study of NS-x

Purpose of NS2, Overview, OTcl: The User Language, Simple Simulation Example, Event Scheduler, Network Components, Packet, Post Simulation: Trace Analysis and Examples, Types of Queue Monitor and Examples,

Unit 3: Basic Scenarios using NS2

Writing OTcl code for following:

1. Scenario for different topologies – star, bus, mesh, ring
2. Wired Scenario for different bandwidth and packet size for 10 nodes for LAN.
3. Scenario for TCP and UDP with proper example.
4. Comparative Graph for any two scenarios.

Books and References:

1. Discrete-Event System Simulation – Jerry Banks, John S. Carson II, Barry L. Nelson,David M. Nicol
2. Simulation Modeling and Analysis – Averill M. Law
3. http://nile.wpi.edu/NS/ -- NS by Example tutorial
4. Network Simulator website and NS2 manual

SOLAPUR UNIVERSITY, SOLAPUR

T.E. (Computer Science and Engineering)

Semester - II

8. Self Learning (Technical) – 2. Network Setup and Management Tools

Examination Scheme

Theory: 50 marks

COURSE OUTCOMES:

At the end of the course, student will be able to

1 Compare between various components of network, select appropriate network topology and setup a computer network.
2 Use the network management tools for monitoring the network performance.
3 Demonstrate different ways of network setup and use of network management tools.

Unit 1: Computer Network Setup

Network Interface Adapters: NIC Functions, Features, Selecting a NIC, Network Connection Devices: Hubs, Switches, VLAN, Layer-3 Switches, Designing a Network: Network design overview, Designing an Internetwork.

Unit 2: Network Management

Network Management Architectures and Applications, Configuration Management and Auto Discovery, Configuration Databases and Reports, Abstract Syntax Notation One (ASN.1)

Unit 3: Network Management Functions

Fault Management, Fault identification and isolation, Event correlation Techniques, Security Management, Host and User Authentication, Key Management.

Unit 4: Management Tools, Systems and Applications

Testing and Monitoring Tools, Integrating Tools, Development Tools, Web-based Enterprise Management.

Books and References:

1. Networking – The Complete Reference by Craig Zacker Tata McGraw Hill (Unit 1)
2. Network Management : Principles and Practices by Subramanian M. MA: Addison – Wesley (2000) (Unit 2,3,4)

SOLAPUR UNIVERSITY, SOLAPUR

T.E. (Computer Science and Engineering)

Semester - II

8. Self Learning (Technical) – 3. Software Licensing and Practices

Examination Scheme

Theory : Marks: 50

COURSE OUTCOMES:

At the end of the course, student will be able to

1 Present software licensing models and practices adopted in software development and distribution.

2 Analyze, compare and choose appropriate software licensing model and strategy for the software developed.

Unit 1: Introduction

Basic Principles of Copyright Law, Contract and Copyright, Open Source Software Licensing, Issues, with Copyrights and Patents, The Open Source Definition, Warranties.

Unit 2: Software Licenses

The MIT License , The BSD License , The Apache License, v1.1 and v2.0, The Academic Free License , Application and Philosophy of MIT and BSD Licenses, GNU General Public License , GNU Lesser General Public License , The Mozilla Public License , Application and Philosophy of GNU GPL and GNU LGPL.

Unit 3: Creative Commons Licenses and Non Open Source Software Licenses

Creative Commons Licenses, Classic Proprietary License, Sun Community Source License, Microsoft Shared Source Initiative.

Unit 4: Legal Impacts of Open Source and Free Software Licensing

Entering Contracts, Statutory Developments Related to Software Contracts, The Self-Enforcing Nature of Open Source and Free Software Licenses, The Global Scope of Open Source and Free Software Licensing, The "Negative Effects" of Open Source and Free Software Licensing, Community Enforcement of Open Source and Free Software Licenses, Compatible and Incompatible Licensing: Multiple and Cross Licensing.

Textbooks:

1. Understanding Open Source and Free Software Licensing - By Andrew M. St. Laurent, Oreily Media.
   (e-Resource available at http://oreilly.com/openbook/osfreesoft/book/index.html)

Reference Books:

1) Intellectual Property and Open Source: A Practical Guide to Protecting Code - By Van Lindberg, O'relliy Media.
2) Essentials of Licensing Intellectual Property - By Alexander I. Poltorak and Paul J. Lerner, John Wiley Publication.