



INFORMATION TECHNOLOGY

Course Structure R-20

SEMESTER V

S. No	Class	Course Code	Name of the Subject	L	T	P	C
1	PC	C65PC1	Software Engineering	3	0	0	3
2	PC	C65PC2	Python Programming	3	0	0	3
3	PC	C65PC3	Compiler Design	3	0	0	3
4	PC	C65PC4	Computer Networks	3	0	0	3
5	PC	C65PC5	Object Oriented Analysis and Design	3	0	0	3
6	OE	C65OE6	Open Elective-I	3	0	0	3
7	PC	C65PC7	Object Oriented Analysis and Design Lab	0	0	3	1.5
8	PC	C65PC8	Python Programming Lab	0	0	3	1.5
9	MC	MC005	MOOCs/Online Course	0	0	0	S
Total Credits							21

Mandatory Course: MOOCs/Online Course

The student should register for any one of the MOOCs course offered by NPTEL, COURSERA, UDEMY, student should submit the completion certificate to clear this course.



INFORMATION TECHNOLOGY

Course Structure R-20

SEMESTER VI

S. No	Class	Course Code	Name of the Subject	L	T	P	C
1	HS	CHSM2	Fundamentals of Management	3	0	0	3
2	PC	C66PC1	Web Technologies	3	0	0	3
3	OE	C66OE2	Open Elective-II	3	0	0	3
4	PE	C66PE3	Professional Elective- I 1) Advanced Databases 2) Network Programming 3) Stack Technologies	3	0	0	3
5	PE	C66PE4	Professional Elective- II 1) Distributed Databases 2) Wireless Networks 3) Mobile Application Development	3	0	0	3
6	HS	CHSE3	Advanced English Communication Skills Lab	0	0	3	2
7	PC	C66PC5	Web Technologies Lab	0	0	2	1
8	PW	C66PW6	Mini Project	0	0	6	3
9	MC	MC006	1. Personality Development/Skill Development/ 2. Technical Events 3. Internships	0	0	0	S
Total Credits							21

Mini Project: Every Student should submit the Mini Project report on any of the “ Emerging Technologies” in the prescribed format through which the student understands the fundamentals through practical application of theoretical concepts.

Mandatory Course : The satisfactory report should be submitted either for 1 or 2 or 3 given below.

1. Personality Development/Skill Development: Student should participate in personality development/communication skills programme, student should submit the completion certificate for clearing this course.

2. Technical Events: The student should participate in any technical event organized by any College /Organization /Industry and submit the participation certificate for clearing this course.

3. Internships: The Student should submit the completion certificate from the respective organization. Where he/she performs their internship.



SOFTWARE ENGINEERING - C65PC1

B. Tech. V Semester

L/T/P/C

3/0/0/3

Course Objectives:

1. To understanding of software process models such as waterfall and evolutionary models.
2. To understanding of software requirements and SRS document.
3. To understanding of different software architectural styles.
4. To understanding of software testing approaches such as unit testing and integration testing
5. To understanding on quality control and how to ensure good quality software.

Course Outcomes:

1. An ability to identify and apply suitable process patterns and process models accordingly.
2. Able to identify requirements, apply requirements engineering process and design system models.
3. Designing Architecture to acquire knowledge of building an application and use of UML diagrams.
4. Able to test software by applying various testing strategies and product metrics to measure the product.
5. Assessing risk factors by formulating risk management and to assess the quality of software.

UNIT I

Introduction to Software Engineering: The evolving role of software, Changing Nature of Software, legacy software, Software myths.

A Generic view of process: Software engineering- A layered technology, a process framework, The Capability Maturity Model Integration (CMMI), Process patterns, process assessment, personal and team process models.

Prescriptive Process models: The waterfall model, incremental process models, evolutionary process models.



UNIT II

Software Requirements: Functional and non-functional requirements, User requirements, System requirements, the software requirements document.

Requirements engineering process: Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management.

System models: Context Models, Behavioral models, Data models, Object models, structured methods, data dictionary.

UNIT III

Design Engineering: Design process and Design quality ,Design concepts, the design model, pattern based software design.

Creating an architectural design: software architecture, Data design, Architectural styles and patterns, Architectural Design, Conceptual model of UML and diagrams in UML.

UNIT IV

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, Black-Box and White-Box testing, Validation testing, System testing, the art of Debugging.

Product metrics: Software Quality, Framework for Product metrics, Metrics for Analysis Model, Metrics for Design Model, Metrics for source code, Metrics for testing, Metrics for maintenance.

Metrics for Process and Products: Software Measurement, Metrics for software quality.

UNIT V

Risk management: Reactive vs. Proactive Risk strategies, software risks, Risk identification, Risk projection, Risk refinement, RMMM, RMMM Plan.

Quality Management: Quality concepts, Software quality assurance, Software Reviews, Formal technical reviews, Statistical Software quality Assurance, Software reliability, The ISO 9000 quality standards.

TEXT BOOKS:

1. Software engineering practitioner's Approach, Roger S Pressman, sixth edition McGraw Hill International Edition.
2. Software Engineering, Ian Somerville, seventh edition, Pearson education.



REFERENCE BOOKS:

1. Software Engineering, A Precise Approach, Pankaj Jalote, Wiley India, 2010.
2. Software Engineering: A Primer, Waman S Jawadekar, Tata McGraw-Hill, 2008,
Fundamentals of Software Engineering, Rajib Mall, PHI, 2005
3. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.
4. Software Engineering 1: Abstraction and modeling, Diner Bjorner, Springer International edition, 2006.
5. Software Engineering 2: Specification of systems and languages, Diner Bjorner, Springer International edition 2006.
6. Software Engineering Foundations, Yingxu Wang, Auerbach Publications, 2008.
7. Software Engineering Principles and Practice, Hans Van Vliet, 3rd edition, John Wiley & Sons Ltd.
8. Software Engineering 3: Domains, Requirements, and Software Design, D. Bjorner, Springer International Edition.



PYTHON PROGRAMMING - C65PC2

B. Tech. V Semester

L/T/P/C

3/0/0/3

Course Objectives:

1. To understand the fundamentals of Python Programming concepts and its applications
2. To improve problem solving skills using control structures and lists.
3. To understand the basics of object- oriented concepts using python.
4. Apply string handling to solve real-time problems.
5. Design and implement programs using functions.

Course Outcomes:

1. Understand and comprehend the basics of python programming.
2. Express different conditional and decision making statements used to develop python applications.
3. Learn and implement various data structures provided by python library including string, list, dictionary and its operations etc
4. Define and demonstrate the use of the built-in functions and better usage of string methods in the development of python programming.
5. Develop real-world applications by using various object oriented programming concepts.

UNIT I

INTRODUCTION TO PYTHON Introduction to Python: Python Identifiers, Keywords, Data types in python: built-in data types, bool data type, sequences, sets. Input and Output statements, Operators: arithmetic operators, assignment operators, comparison operators, logical operators, identity operators, membership operators, bitwise operators.

UNIT II

CONTROL STRUCTURES Conditional Control structures: Conditional blocks using if statement, if-else statement, else if statement, Range function. Loops: for loops, Nested for loop, while loop, pass, continue, break statements.



UNIT III

LIST, TUPLES ,DICTIONARY AND ARRAYS Creating List, List manipulation – index(), append(), insert(), copy(), extend(), count(), remove(), pop(), reverse(), sort(), len(), nested list. Creating a tuple, accessing a tuple element, basic operations on tuples, tuples manipulations – len(),min(), max(), count(),index(), sorted().Creation of dictionary, operations on dictionaries, dictionaries methods Array: creating an array, importing array module, indexing and slicing. Processing the arrays, arrays using numpy, array creation using numpy, transpose, addition and multiplication of matrices.

UNIT IV

STRINGS AND FUNCTIONS Creating a string, methods – length(), indexing(), slicing(), repeating(), concatenation(), comparing(), remove(), removing spaces, finding substring, inserting a sub string in to a string, finding number of characters and words. Functions: Defining a function, calling a function, returning multiple values from a function, functions are first class objects, formal and actual arguments, positional arguments, recursive functions, Powerful Lamda function in python.

UNIT V

NUMPY AND PANDAS Introduction, Creating nd arrays, Data Types for nd Arrays, Indexing and Slicing, Mathematical and statistical methods, Methods for Boolean Arrays, sorting. Introduction to PANDAS, series, Data Frame, Index objects, Essential Functionality, Handling Missing Data.

TEXT BOOKS:

1. R Nageswara Rao, –Core Python Programming||, Dreamtech Press, 2nd Edition, 2017.
2. Dusty Philips, –Python 3 Object Oriented Programming||, PACKT Publishing, 2nd Edition, 2015.
3. Wes McKinney, –Python for Data Analysis|| Data Wrangling with Pandas, Numpy, and IPython. 2nd Edition O'REILLY,2017.



TKR COLLEGE OF ENGINEERING & TECHNOLOGY

(Autonomous, Accredited by NAAC with A Grade)

B. Tech INFORMATION TECHNOLOGY – R20



REFERENCE BOOKS:

1. Michael H. Goldwasser, David Letscher, –Object Oriented Programming in PythonII, Prentice Hall, 1st Edition, 2007.



COMPILER DESIGN- C65PC3

B. Tech. V Semester

L/T/P/C

3/0/0/3

Course Objectives:

1. To understand the various phases in the design of a compiler.
2. To understand the design of top-down and bottom-up parsers.
3. To understand syntax directed translation schemes.
4. To introduce lex and yacc tools.
5. To learn to develop algorithms to generate code for a target machine.

Course Outcomes:

1. Ability to design, develops, and implements a compiler for any language.
2. Able to use lex and yacc tools for developing a scanner and a parser.
3. Able to design and implement LL and LR parsers.
4. Able to design algorithms to perform code optimization in order to improve the performance of a program in terms of space and time complexity.
5. Ability to design algorithms to generate machine code

UNIT - I

Introduction: Language Processors, the structure of a compiler, the science of building a compiler, programming language basics.

Lexical Analysis: The Role of the Lexical Analyzer, Input Buffering, Recognition of Tokens, The Lexical-Analyzer Generator Lex, Finite Automata, From Regular Expressions to Automata, Design of a Lexical-Analyzer Generator, Optimization of DFA-Based Pattern Matchers.

UNIT - II

Syntax Analysis: Introduction, Context-Free Grammars, Writing a Grammar, Top-Down Parsing, Bottom-Up Parsing, Introduction to LR Parsing: Simple LR, More Powerful LR Parsers, Using Ambiguous Grammars, Parser Generators.



UNIT – III

Syntax-Directed Translation: Syntax-Directed Definitions, Evaluation Orders for SDD's, Applications of Syntax-Directed Translation, Syntax-Directed Translation Schemes, and Implementing L-Attributed SDD's.

Intermediate-Code Generation: Variants of Syntax Trees, Three-Address Code, Types and Declarations, Type Checking, Control Flow, Back patching, Switch-Statements, Intermediate Code for Procedures.

UNIT – IV

Run-Time Environments: Storage organization, Stack Allocation of Space, Access to Nonlocal Data on the Stack, Heap Management, Introduction to Garbage Collection, Introduction to Trace-Based Collection.

Code Generation: Issues in the Design of a Code Generator, The Target Language, Addresses in the Target Code, Basic Blocks and Flow Graphs, Optimization of Basic Blocks, A Simple Code Generator, Peephole Optimization, Register Allocation and Assignment, Dynamic Programming Code-Generation.

UNIT - V

Machine-Independent Optimizations: The Principal Sources of Optimization, Introduction to Data-Flow Analysis, Foundations of Data-Flow Analysis, Constant Propagation, Partial Redundancy Elimination, Loops in Flow Graphs

TEXT BOOK:

1. Compilers: Principles, Techniques and Tools, Second Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, Pearson.

REFERENCE BOOKS:

1. Compiler Construction-Principles and Practice, Kenneth C Louden, Cengage Learning.
2. Modern compiler implementation in C, Andrew W Appel, Revised edition, Cambridge University Press.
3. The Theory and Practice of Compiler writing, J. P. Tremblay and P. G. Sorenson, TMH



COMPUTER NETWORKS - C65PC4

B. Tech. V Semester

L/T/P/C

3/0/0/3

Course Objectives:

1. To introduce the fundamental various types of computer networks.
2. To demonstrate the TCP/IP and OSI models with merits and demerits.
3. To explore the various layers of OSI Model.
4. To introduce UDP and TCP Models.

Course Outcomes:

1. Able to understand and explore the basics of data communication.
2. Able to understand data link layer with transmission error to provide a well defined interface to the network layer.
3. Classify the routing protocols and analyzes how to assign the IP addresses for a given network.
4. Able to understand to perform end to end services in the transport layer.
5. Ability to access the global information about services on the Internet.

UNIT – I

Introduction: Data Communications, Networks, The Internet, Protocols and Standards, Layered Tasks, The OSI model, Layers in the OSI Model, TCP/IP, Addressing, **Physical layer:** Transmission modes, Multiplexing, Transmission Media, Switching - Circuit Switched Networks, Datagram Networks, Virtual Circuit Networks.

UNIT – II

Data link layer: Introduction, Error Detection and Correction, Framing, Flow and Error Control, Noiseless Channels, Noisy Channels, HDLC, Point to Point Protocols.

Multiple Access: Random Access, ALOHA, CSMA, CSMA/CD, CSMA/CA, Controlled access, Channelization.



UNIT – III

Network Layer: IPV4 Addresses, IPV6 Addresses, Internetworking, IPV4, IPV6, Transition From IPV4 To IPV6, Address Mapping, ICMP, IGMP, Delivery, Forwarding, Unicast Routing Protocols, Intra and Interdomain Routing, Unicast and Multicast Routing Protocols, Connecting Devices-Passive Hubs, Repeaters, Bridge, Routers, Gateway.

UNIT – IV

Transport Layer: Process to Process Delivery, UDP and TCP protocols, SCTP. **Congestion Control and Quality of Service:** Data Traffic, Congestion, Congestion Control, QoS, Techniques to improve QoS, Integrated Services, Differentiated Services, QoS in Switched Networks.

UNIT – V

Application Layer: Name Space, DNS, DNS in Internet, TELNET, Electronic Mail, File Transfer, Architecture of WWW, HTTP, SNMP.

Network Security: Introduction, Security goals, Attacks, Services and Techniques, Confidentiality, Network Layer Security, Application Layer Security, Firewalls.

TEXT BOOKS:

1. Data Communications and Networking – Behrouz A. Forouzan, 5th Edition Mc Graw Hill Education, 2006.
2. Computer Networks — Andrew S Tanenbaum, 4th Edition, Pearson Education.
3. Computer Networking: A Top-Down Approach Featuring the Internet, James F. Kurose, K. W. Ross, 3rd Edition, Pearson Education.

REFERENCES:

1. Data communications and Networks by William Stallings
2. Data communication and Networks – Bhusan Trivedi, Oxford university press 2016.
3. An Engineering Approach to Computer Networks-S. Keshav, 2nd Edition, Pearson Education.
4. Understanding Communications and Networks, 3rd Edition, W. A. Shay, Cengage Learning



OBJECT ORIENTED ANALYSIS AND DESIGN - C65PC5

B. Tech. V Semester

L/T/P/C

3/0/0/3

Course Objectives:

1. Concisely define the following key terms: class, object, state, behavior, object class, class diagram, object diagram, operation, encapsulation, constructor operation, query operation, update operation, scope operation, association, association role, multiplicity, association class, abstract class, concrete class, class-scope attribute, abstract operation, method, polymorphism, overriding, multiple classification, aggregation, and composition.
2. State the advantages of object-oriented modeling vis-à-vis structured approaches.
3. Model a real-world application by using a UML class diagram.
4. Recognize when to use generalization, aggregation, and composition relationships.
5. Specify different types of business rules in a class diagram.

Course Outcomes:

Graduate can able to take up the case studies and model it in different views with respect user requirement such as use case, logical, component and deployment and etc, and preparation of document of the project for the unified Library application.

UNIT– I

Introduction to UML: Importance of modelling, principles of modelling, object oriented modelling, conceptual model of the UML, Architecture, Software Development Life Cycle.

UNIT– II

Basic Structural Modelling: Classes, Relationships, common Mechanisms, and diagrams. Advanced Structural Modelling: Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages. Class & Object Diagrams: Terms, concepts, modelling techniques for Class & Object Diagrams.



UNIT– III

Basic Behavioural Modelling-I: Interactions, Interaction diagrams. Basic Behavioural Modelling-II: Use cases, Use case Diagrams, Activity Diagrams.

UNIT– IV

Advanced Behavioural Modelling: Events and signals, state machines, processes and Threads, time and space, state chart diagrams Architectural Modelling: Component, Deployment, Component diagrams and Deployment diagrams.

UNIT– V

Patterns and Frameworks, Artificer Diagrams. Case Study: The Unified library application, **ATM application.**

TEXT BOOKS:

1. Grady Booch, James Rumbaugh, Ivar Jacobson : The Unified Modeling Language User Guide, Pearson Education 2nd Edition
2. Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado: UML 2 Toolkit, WILEY-Dreamtech India Pvt. Ltd.

REFERENCE BOOKS:

1. Meilir Page-Jones: Fundamentals of Object Oriented Design in UML Pearson Education.
2. Pascal Rogues: Modeling Software Systems Using UML2, WILEY- Dreamtech India Pvt. Ltd.
3. AtulKahate: Object Oriented Analysis & Design, The McG raw Hills Companies.
4. Mark Priestley: Practical Object-Oriented Design with UML, TMH.
5. Appling UML and Patterns: An introduction to Object — Oriented
6. Analysis and Design and Unified Process, Craig Larman, Pearson Education.
7. Object-Oriented Analysis and Design with the Unified Process by John W. Satzinger, Robert B Jackson and Stephen D Burd, Cengage Learning.
8. UML and C++, R.C.Lee, and W.M.Tepfenhart, PHI.
9. Object Oriented Analysis, Design and Implementation, B.Dathan. S.Ramnath, Universities Press.
10. Design with UML and Java, K.Barclay, J.Savage, Elsevier.



OBJECT ORIENTED ANALYSIS AND DESIGN LAB - C65PC7

B. Tech. V Semester

L/T/P/C
0/0/3/1.5

Course Objectives:

1. Able to identify the requirements specification for an intended software system.
2. Demonstrate how to draw the UML diagrams for the given specification.
3. Illustrate to map the design properly to code.

Course Outcomes:

Upon completion of this course, the students will be able to:

1. Perform Object Oriented analysis and design for a given problem specification.
2. Identify and map basic software requirements in UML mapping.
3. Improve the software quality using design patterns and to explain the rationale behind applying specific design.

A) Students have to draw the following diagrams using UML for an ATM system whose description is given below. UML diagrams to be developed are:

- Use Case Diagram.
- Class Diagram.
- Sequence Diagram.
- Collaboration Diagram.
- State Diagram
- Activity Diagram.
- Component Diagram
- Deployment Diagram.
- Test Design.
- Description for an ATM System



The software to be designed will control a simulated automated teller machine (ATM) having a magnetic stripe reader for reading an ATM card, a customer console (keyboard and display) for interaction with the customer, a slot for depositing envelopes, a dispenser for cash (in multiples of Rs. 100, Rs. 500 and Rs. 1000), a printer for printing customer receipts, and a key-operated switch to allow an operator to start or stop the machine. The ATM will communicate with the bank's computer over an appropriate communication link. (The software on the latter is not part of the requirements for this problem.) The ATM will service one customer at a time. A customer will be required to insert an ATM card and enter a personal identification number (PIN) – both of which will be sent to the bank for validation as part of each transaction.

The customer will then be able to perform one or more transactions. The card will be retained in the machine until the customer indicates that he/she desires no further transactions, at which point it will be returned – except as noted below.

The ATM must be able to provide the following services to the customer

- A customer must be able to make a cash withdrawal from any suitable account linked to the card, in multiples of Rs. 100 or Rs. 500 or Rs. 1000. Approval must be obtained from the bank before cash is dispensed.
- A customer must be able to make a deposit to any account linked to the card, consisting of cash and/or checks in an envelope. The customer will enter the amount of the deposit into the ATM, subject to manual verification when the envelope is removed from the machine by an operator. Approval must be obtained from the bank before physically accepting the envelope.
- A customer must be able to make a transfer of money between any two accounts linked to the card.
- A customer must be able to make a balance inquiry of any account linked to the card.



- A customer must be able to abort a transaction in progress by pressing the Cancel key instead of responding to a request from the machine.

The ATM will communicate each transaction to the bank and obtain verification that it was allowed by the bank. Ordinarily, a transaction will be considered complete by the bank once it has been approved. In the case of a deposit, a second message will be sent to the bank indicating that the customer has deposited the envelope. (If the customer fails to deposit the envelope Within the timeout period or presses cancel instead, no second message will be sent to the bank and the deposit will not be credited to the customer.) If the bank determines that the customer's PIN is invalid, the customer will be required to re-enter the PIN before a transaction can proceed. If the customer is unable to successfully enter the PIN after three tries, the card will be permanently retained by the machine, and the customer will have to contact the bank to get it back. If a transaction fails for any reason other than an invalid PIN, the ATM will display an explanation of the problem, and will then ask the customer whether he/she wants to do another transaction.

The ATM will provide the customer with a printed receipt for each successful transaction

The ATM will have a key-operated switch that will allow an operator to start and stop the servicing of customers. After turning the switch to the non|| position, the operator will be required to verify and enter the total cash on hand. The machine can only be turned off when it is not servicing a customer. When the switch is moved to the hof|| position, the machine will shut down, so that the operator may remove deposit envelopes and reload the machine with cash, blank receipts, etc.

B) Students have to draw the 9 types of UML diagrams for Library Management System.



PYTHON PROGRAMMING LAB - C65PC8

B. Tech. V Semester

L/T/P/C

0/0/3/1.5

Course Objectives:

To Write and execute the programs based on operators, functions, simple data structures, basic packages using python programming constructs.

Course Outcomes:

After completion of course the students will able to

1. Implement the fundamental programming elements: operators, statements, conditional and control flow statements.
2. Use predefined functions and build functions.
3. Use python modules and implement data structure to solve various computing problems.

List of Programs:

1. Write a python program to print —Hello Worldl.
2. Write a python program to demonstrate different number data types in python.
3. Write a program to perform different Arithmetic Operations on numbers in Python.
4. Write a program to create, concatenate and print a string and accessing sub-string from a given string.
5. Write a python script to print the current date?
6. Write a program to create, append, and remove lists in python.
7. Write a program to demonstrate working with tuples in python
8. Write a program to demonstrate working with dictionaries in python.
9. Write a python program to find largest of three numbers.
10. Write a Python program to convert temperatures to and from Celsius, Fahrenheit.



WEB TECHNOLOGIES – (C66PC1)

B. Tech. VI Semester

L/T/P/C

3/0/0/3

Course Objectives:

1. To design Static web pages using HTML.
2. To introduce Client Side scripting with JavaScript and AJAX.
3. To introduce PHP language for Server Side Scripting.
4. To introduce XML and XML data with Java
5. To introduce server side programming with Java Servlets and JSP.

Course Outcomes:

1. Able to design a static web page using forms and frames in HTML.
2. Able to validate client side scripting using onClick (), onSubmit(), onChange() events in JavaScript.
3. Able to design a dynamic web page using PHP.
4. Able to construct a validation page which connects to a data base given and able to perform the DML functionalities by using mysqli_connect() ,mysqli_query(), mysqli_fetch_array(), mysqli_close() in PHP.
5. Analyze how to develop a well formed and valid xml document by using DTDs and Schemas which allows the validation of text elements.
6. Able to write server side program by using servlets for given problem and able to develop a connection between both the ends by doGet() and doPost() methods.& Develop JSP applications implementing Session Management and Database Connectivity.

UNIT-I

Introduction to HTML: HTML basic tags, Elements, Attributes, list, table, image, text links, forms, frames, Cascading style sheets, Simple AJAX application.

UNIT-II

Introduction to Java script: Java script language- declaring variables, scope of variables, operators, loops, functions, Java script objects, event handlers (on click, on submit etc.), Document Object Model.

UNIT-III

Introduction to PHP: Declaring variables, data types, arrays, strings, operators, expressions, Arrays, control structures, functions, Reading data from web form controls



like textboxes, radio buttons, list etc., Connecting to database(My SQL as reference), executing simple queries, handling results, Handling sessions and cookies.

UNIT-IV

XML: Introduction to XML, Defining XML tags, attributes and values, Document Type Definition, XML Schemas, Document Object Model, XHTML

Parsing XML Data- DOM and SAX Parsers in java.

UNIT-V

Introduction to Servlets: Common Gateway Interface (CGI), Lifecycle of a Servlet, deploying a servlet, The Servlet API, Using Cookies and Sessions ,connecting to a database using JDBC.

Introduction to JSP: The Anatomy of a JSP Page, JSP Processing, Declarations, Directives, Expressions, Code Snippets ,implicit objects, Using Beans in JSP Pages.

TEXTBOOKS:

1. Web Technologies, Uttam K Roy,Oxford University Press
2. The Complete Reference PHP–Steven Holzner, Tata McGraw-Hill.

REFERENCEBOOKS:

1. Web Programming, building internet applications, Chris Bates 2nd edition, Wiley Dreamtech
2. Java Server Pages– Hans Bergsten, SPDO“ Reilly
3. Java Script, D. Flanagan, O“Reilly, SPD.
4. Beginning Web Programming- Jon Duckett WROX.
5. Programming World Wide Web, R. W. Sebesta, Fourth Edition, Pearson.
6. Internet and World Wide Web– How to program, Dietel and Nieto, Pearson.



ADVANCED DATABASES - C66PE3A

B. Tech. VI Semester

L/T/P/C

3/0/0/3

Course Objectives:

1. Be able to design high-quality relational databases and database applications.
2. Have developed skills in advanced visual& conceptual modeling and database design.
3. Be able to translate complex conceptual data models into logical and physical data Base designs.
4. Have developed an appreciation of emerging database trends as they apply to semi-structured data, the internet, and object-oriented databases.

Course Outcomes:

1. Understand theoretical and practical aspects of distributed database systems.
2. Study and identify various issues related to the development of distributed database system.
3. Design and implement advanced queries using Structured Query Language
4. Design, construct and maintain a database and various database objects using procedural language constructs, forms and reports to solve problems
5. Implement and maintain database security mechanisms

UNIT I

Database system Architecture: computer-system architecture, and describes the influence of the underlying computer system on the database system.

Parallel databases: explores a variety of parallelization techniques, including I/O parallelism, interquery and intraquery parallelism, and interoperation and intra operation parallelism. Query optimization, Design of parallel Systems, Parallelism on Multi Core Processors.

UNIT II

Object Based Databases: Complex data types, Structured types and Inheritance in SQL, Table Inheritance, Array and Multiset types in SQL, Object-Identity and Reference types in SQL, Implementing O-R features, Persistent Programming Languages, Object-Oriented versus Object- Relational.

XML Databases: Structure of XML data, XML document schema, Querying and Transformation, API to XML, Storage of XML data, XML applications



UNIT III

Information Retrieval: Relevance ranking using Terms, Relevance using Hyperlinks, Synonyms, Homonyms and Ontologies, Indexing of Documents, Measuring Retrieval Effectiveness, Crawling and Indexing the web, Beyond ranking of pages, Directories and Categories

UNIT IV

Advance application Development: Performance Tuning, Performance Benchmarks, Other Issues in Application Development, Standardization

Spatial and Temporal data Mobility: Motivation, Time in Databases, Spatial and Geographical Data Geographic Data, Multimedia Databases, Mobility and Personal databases

Advance Transaction Processing: Transaction-Processing Monitors, Transactional Workflows, E-commerce, Main-memory databases, Real-Time Transaction Systems, Long-Duration Transactions.

UNIT V

Introduction to database security issues, Discretionary access control based on granting and revoking privileges; Mandatory access control and role based access control for multilevel security. SQL Injection, Introduction to statistical Database Security, Introduction to flow control, Encryption and Public Key Infrastructures, Privacy issues and preservation, challenges to maintaining database security.

Text Books:

1. Database Systems concepts 6th edition silberschatz-korth-surdarshan Tata Mc Graw Hill Publications (Indian Edition)
2. Fundamentals of Database systems seventh edition Pearson Publications by Ramez Elmasri, Shamakanth.B. Navathe



NETWORK PROGRAMMING - C66PE3B

B. Tech. VI Semester

L/T/P/C

3/0/0/3

Course Objectives:

1. Demonstrate mastery of main protocols comprising the Internet.
2. Develop skills in network programming techniques.
3. Implement network services that communicate through the Internet.
4. Apply the client-server model in networking applications.
5. Practice networking commands available through the operating systems.

Course Outcomes:

1. To write socket API based programs
2. To design and implement client-server applications using TCP and UDP sockets
3. To analyze network programs

UNIT –I

Introduction to Network Programming: OSI model, Unix standards, TCP and UDP & TCP connection establishment and Format, Buffer sizes and limitation, standard internet services, Protocol usage by common internet application.

Sockets: Address structures, value – result arguments, Byte ordering and manipulation function and related functions Elementary TCP sockets – Socket, connect, bind, listen, accept, fork and exec function, concurrent servers, Close function and related function.

UNIT-II

TCP client server: Introduction, TCP Echo server functions, Normal startup, terminate and signal handling server process termination, Crashing and Rebooting of server host shutdown of server host.

Elementary UDP sockets: Introduction UDP Echo server function, lost datagram, summary of UDP example, Lack of flow control with UDP, determining outgoing interface with UDP.



I/O Multiplexing: I/O Models, select function, Batch input, shutdown function, poll function, TCP Echo server,

UNIT-III

Socket options: getsockopt and setsockopt functions. Socket states, Generic socket option IPV6 socket option ICMPV6 socket option IPV6 socket option and TCP socket options.

Advanced I/O Functions-Introduction, Socket Timeouts, recv and send Functions, readv and writev Functions, recvmsg and sendmsg Functions, Ancillary Data, Sockets and Standard I/O, T/TCP: TCP for Transactions.

UNIT-IV

Elementary name and Address conversions: DNS, gethost by Name function, Resolver option, Function and IPV6 support, uname function, other networking information.

Daemon Processes and inetd Superserver – Introduction, syslogd Daemon, syslog Function, daemon_init Function, inetd Daemon, daemon_inetd Function

Broadcasting- Introduction, Broadcast Addresses, Unicast versus Broadcast, dg_cli Function Using Broadcasting, Race Conditions

Multicasting- Introduction, Multicast Addresses, Multicasting versus Broadcasting on A LAN, Multicasting on a WAN, Multicast Socket Options, mcast_join and Related Functions, dg_cli Function Using Multicasting, Receiving Mbone Session Announcements, Sending and Receiving, SNTP: Simple Network Time Protocol, SNTP (Continued)

UNIT-V

Raw Sockets- Introduction, Raw Socket Creation, Raw Socket Output, Raw Socket Input, Ping Program, Traceroute Program, An ICMP Message Daemon, Datalink Access- Introduction, BPF: BSD Packet Filter, DLPI: Data Link Provider Interface, Linux: **SOCK_PACKET**, **libpcap**: Packet Capture Library, Examining the UDP Checksum Field. Remote Login: Terminal line disciplines, Pseudo-Terminals, Terminal modes, Control Terminals, rlogin Overview, RPC Transparency Issues.



TKR COLLEGE OF ENGINEERING & TECHNOLOGY

(Autonomous, Accredited by NAAC with „A Grade)

B. Tech INFORMATION TECHNOLOGY – R20



TEXT BOOKS:

1. UNIX Network Programming, by W. Richard Stevens, Bill Fenner, Andrew M. Rudoff, Pearson Education
2. UNIX Network Programming, 1st Edition, - W. Richard Stevens. PHI.

REFERENCES:

1. UNIX Systems Programming using C++ T CHAN, PHI.
2. UNIX for Programmers and Users, 3rd Edition Graham GLASS, King abls, Pearson Education
3. Advanced UNIX Programming 2nd Edition M. J. ROCHKIND, Pearson Education



STACK TECHNOLOGIES - C66PE3C

B. Tech. VI Semester

L/T/P/C

3/0/0/3

Course Objectives:

1. Translate user requirements into the overall architecture and implementation of new systems and manage project and coordinate with the client
2. writing optimized front end code HTML and Java Script
3. Design and implementation of Robust and scalable front end applications.

Course Outcomes:

1. Enumerate the Basic concepts of web & Markup Languages.
2. Develop web Applications using scripting Languages & Frameworks
3. Make use of Express JS and and Node JS Frameworks
4. Illustrate the uses of web services concepts like react js

UNIT I

Hibernate - An Introduction to Hibernate 3 , Integrating and Configuring Hibernate, Building a Simple Application ,The Persistence Life Cycle, An Overview of Mapping, Mapping with Annotations, Creating Mappings with Hibernate XML Files , Using the Session , Searches and Queries, Advanced Queries Using Criteria .

UNIT II

Spring - Springing into Action , Wiring beans, Advanced wiring(3.3,3.4), Building Spring web applications, Hitting the database with Spring and JDBC, Persisting data with object-relational mapping, Creating REST APIs with Spring MVC

UNIT III

Spring Boot - Spring Boot Introduction, Spring-boot basics, Spring MVC, Data Access

UNIT IV

React JS - Introduction to Meet React, <Hello World />: our first component , Data and data flow in React, Rendering and lifecycle methods in React, Working with forms in React, Integrating third-party libraries with React.



UNIT V

Node.js: Introduction, Advantages, Node.js Process Model, Node JS Modules. Express. Js: Introduction to Express Framework, Introduction to Nodejs, What is Nodejs, Getting started with Express, your first Express App, Express Routing, Implementing MVC in Express.

TEXT BOOKS:

1. Hibernate in Action CHRISTIAN BAUER GAVIN KING, MANNING
2. Spring Boot in Action Craig Walls, MANNING
3. Spring Micro services in Action JOHN CARNELL, MANNING
4. React JS in Action Mark Tielens Thomas, MANNING.

REFERENCE BOOKS:

1. Express.JS Guide, The Comprehensive Book on Express.js, Azat Mardan, Lean Publishing.
2. Java Script & jquery the missing manual, David sawyer mcfarland, O'Reilly.



DISTRIBUTED DATABASES - C66PE4A

B. Tech. VI Semester

L/T/P/C

3/0/0/3

Course Objectives:

1. The purpose of the course is to enrich the previous knowledge of database systems and exposing the need for distributed database technology to confront with the deficiencies of the centralized database systems.
2. Introduce basic principles and implementation techniques of distributed database systems.
3. Equip students with principles and knowledge of parallel and object-oriented databases.
4. Topics include distributed DBMS architecture and design; query processing and optimization; distributed transaction management and reliability; parallel and object database management systems.

Course Outcomes:

1. Understand theoretical and practical aspects of distributed database systems.
2. Study and identify various issues related to the development of distributed database system.
3. Understand the design aspects of object-oriented database system and related development.

UNIT –I

Introduction: Distributed Data Processing, Distributed Database System, Promises of DDBSs, Problem areas.

Distributed DBMS Architecture: Architectural Models for Distributed DBMS, DDMBS Architecture.

Distributed Database Design: Alternative Design Strategies, Distribution Design issues, Fragmentation, Allocation.

UNIT-II

Query processing and decomposition: Query processing objectives, characterization of query processors, layers of query processing, query decomposition, localization of distributed data.

Distributed query Optimization: Query optimization, centralized query optimization, distributed query optimization algorithms.



UNIT-III

Transaction Management: Definition, properties of transaction, types of transactions, distributed concurrency control: serializability, concurrency control mechanisms & algorithms, time - stamped & optimistic concurrency control Algorithms, deadlock Management.

UNIT- IV

Distributed DBMS Reliability: Reliability concepts and measures, fault-tolerance in distributed systems, failures in Distributed DBMS, local & distributed reliability protocols, site failures and network partitioning.

Parallel Database Systems: Parallel database system architectures, parallel data placement, parallel query processing, load balancing, database clusters.

UNIT-V

Distributed object Database Management Systems: Fundamental object concepts and models, object distributed design, architectural issues, object management, distributed object storage, object query Processing.

Object Oriented Data Model: Inheritance, object identity, persistent programming languages, persistence of objects, comparison OODBMS and ORDBMS

TEXT BOOKS:

1. M. Tamer OZSU and Patuck Valduriez: Principles of Distributed Database Systems, Pearson Edn. Asia, 2001.
2. Stefano Ceri and Giuseppe Pelagatti: Distributed Databases, McGraw Hill.

REFERENCE BOOKS:

1. Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer Widom: “Database Systems: The Complete Book”, Second Edition, Pearson International Edition



WIRELESS NETWORKS - C66PE4B

B. Tech. VI Semester

L/T/P/C

3/0/0/3

Course Objectives:

1. To learn about the issues and challenges in the design of wireless ad hoc networks.
2. To understand the working of MAC and Routing Protocols for ad hoc and sensor networks
3. To learn about the Transport Layer protocols and their QoS for ad hoc and sensor networks.
4. To understand various security issues in ad hoc and sensor networks and the corresponding solutions.

Course Outcomes:

On successful completion of the course the students will be able to

1. List and explain the various issues and applications of Ad hoc wireless networks.
2. Classify and Explain the working of MAC protocols for Ad-hoc wireless networks
3. Discuss the issues in designing routing protocols and working of Table-Driven Routing protocols.
4. Compare and contrast the working of various On-Demand Routing protocols.
5. Analyze the challenges in designing Transport layer Protocols for Ad-hoc networks, Compare and contrast the working of Transport protocols.
6. Identify the issues in designing Security Protocols for Ad-hoc networks focusing on the working performance of various security protocols.

UNIT – I

Introduction: Cellular and Ad Hoc Wireless networks, Applications of Ad Hoc wireless networks; Issues in Ad hoc wireless networks: Medium access scheme, routing, multicasting, transport Layer Protocols, Pricing Scheme, Quality of service positioning, Self-organization, Security, Addressing and Service Discovery, Energy Management, Scalability, Deployment considerations.

UNIT – II

MAC protocols: MAC Protocols for Ad hoc wireless Networks: Introduction, Issues in designing a MAC protocol for Ad hoc wireless Networks, Design goals of a MAC protocol for Ad-hoc wireless Networks, Classification of MAC Protocols, Contention based protocols with reservation mechanisms: D-PRMA, CATA, SRMA/PA, FPRP, HRMA



UNIT – III

Routing Protocols: Routing protocols for Ad-hoc Wireless Networks: Introduction, Issues in Designing a Routing Protocol for Ad-hoc Wireless Networks, Classification of Routing Protocols. Table driven routing Protocols: DSDV, WRP, On-Demand Routing Protocols: Dynamic source Routing Protocol DSR, AODV, TORA, LAR, ABR.

UNIT – IV

TRANSPORT LAYER: Transport Layer Protocols for Ad-hoc wireless Networks: Introduction, Issues in Designing a Transport Layer Protocol for Ad-hoc wireless Networks, Design Goals of a Transport Layer Protocol for Adhoc wireless Networks, Classification of Transport Layer Solutions, TCP over Ad-hoc wireless Networks: Feedback-Based TCP, TCP with Explicit Failure Notification, TCP-BUS, Ad-hoc TCP, Split TCP.

UNIT – V

Security: Security in wireless Ad hoc wireless Networks, Network security Requirements, Issues & Challenges in Security Provisioning, Network security Attacks, Key Management: Symmetric and Asymmetric key Algorithms, key Management Approaches, key management in Ad-hoc Wireless Networks: Secure routing in Ad hoc wireless Networks: Requirements, SAR protocol, Security- Aware AODV protocol.

TEXT BOOK:

1. Ad hoc Wireless Networks– C. Siva Ram Murthy & B.S. Manoj, 2nd Edition, Pearson Education, 2005.

REFERENCE BOOKS:

1. Ad hoc Wireless Networks – OzanK. Tonguz and Gianguigi Ferrari, JohnWiley,2006.
2. Ad hoc Wireless Networking – Xiuzhen Cheng, Xiao Hung, Ding-Zhu Du, Kluwer Academic Publishers,2004.
3. Adhoc Mobile Wireless Networks - C.K. Toh, Protocols and Systems, Prentice-Hall PTR,2000.



MOBILE APPLICATION DEVELOPMENT - C66PE4C

B. Tech. VI Semester

L/T/P/C

3/0/0/3

Course Objectives:

1. To demonstrate their understanding of the fundamentals of Android operating systems
2. To improve their skills of using Android software development tools
3. To demonstrate their ability to develop software with reasonable complexity on mobile platform
4. To demonstrate their ability to deploy software to mobile devices
5. To demonstrate their ability to debug programs running on mobile devices

Course Outcomes:

1. Student understands the working of Android OS Practically.
2. Student will be able to develop Android user interfaces
3. Student will be able to develop, deploy and maintain the Android Applications.

UNIT – I

Introduction to Android Operating System: Android OS design and Features – Android development framework, SDK features, Installing and running applications on Android Studio, Creating AVDs, Types of Android applications, Best practices in Android programming, Android tools.

Android application components – Android Manifest file, Externalizing resources like values, themes, layouts, Menus etc, Resources for different devices and languages, Runtime Configuration Changes

Android Application Lifecycle – Activities, Activity lifecycle, activity states, monitoring state changes

UNIT - II

Android User Interface: Measurements – Device and pixel density independent measuring

UNIT – s Layouts – Linear, Relative, Grid and Table Layouts User Interface (UI) Components

– Editable and non-editable TextViews, Buttons, Radio and Toggle Buttons, Checkboxes, Spinners, Dialog and pickers

Event Handling – Handling clicks or changes of various UI components



Fragments – Creating fragments, Lifecycle of fragments, Fragment states, Adding fragments to Activity, adding, removing and replacing fragments with fragment transactions, interfacing between fragments and Activities, Multi-screen Activities

UNIT - III

Intents and Broadcasts: Intent – Using intents to launch Activities, Explicitly starting new Activity, Implicit Intents, Passing data to Intents, Getting results from Activities, Native Actions, using Intent to dial a number or to send SMS

Broadcast Receivers – Using Intent filters to service implicit Intents, Resolving Intent filters, finding and using Intents received within an Activity

Notifications – Creating and Displaying notifications, Displaying Toasts

UNIT - IV

Persistent Storage: Files – Using application specific folders and files, creating files, reading data from files, listing contents of a directory Shared Preferences – Creating shared preferences, saving and retrieving data using Shared Preference

Database – Introduction to SQLite database, creating and opening a database, creating tables, inserting retrieving and etindelg data

UNIT – V

iOS Programming: Apps for a Mobile Platform, iOS Benefits, iOS App Development Essentials, The Application Model., Examining an Objective-C Program., Defining Classes, Using Classes, Objects, Methods, and Variables, Managing Memory, Handling Exceptions, Organizing Program Files, Analyzing Objective-C’s Object-Orientation Capabilities.

TEXT BOOKS:

1. Professional Android 4 Application Development, Reto Meier, Wiley India, (Wrox), 2012
2. Android Application Development for Java Programmers, James C Sheusi, Cengage Learning, 2013

REFERENCES:

1. Beginning Android 4 Application Development, Wei-Meng Lee, Wiley India (Wrox), 2013



WEB TECHNOLOGIES LAB - C66PC5

B. Tech. VI Semester

L/T/P/C

0/0/2/1

Course Objective:

Develop an ability to design and implement static and dynamic website, choose best technologies for solving web client/server problems, use appropriate client-side or Server-side applications

Course Outcomes:

Upon successful completion of this course, the students will be able to:

1. Create web pages using HTML and Cascading Styles sheets
2. Create dynamic web pages using JavaScript & Analyze a web page and identify its elements and attributes
3. Build web applications using PHP
4. Understand, analyze and apply the role of languages like HTML, CSS, XML, JavaScript, PHP and protocols in the workings of the web and web applications
5. Create XML documents and XML Schema.
6. Create web based applications using Servlets &JSP and establish a database connectivity using JDBC.

List of Programs:

1. Write a HTML code to design a simple timetable using table tag.
2. Write a HTML code to design a static college website that holds the complete information about the all departments.
3. Write a JavaScript to design a simple calculator to perform the following operations: sum, product, difference and quotient.
4. Write a JavaScript that calculates the squares and cubes of the numbers from 0 to 10 and outputs HTML text that displays the resulting values in an HTML table format.
5. Write a JavaScript code that displays text “TEXT-GROWING” with increasing font size in the interval of 100ms in RED COLOR, when the font size reaches 50pt it displays “TEXTSHRINKING” in BLUE color. Then the font size decreases to 5pt.
6. Design an XML document to store information about a student in TKR engineering college affiliated to JNTUH. The information must include USN, Name, and Name of the College, Branch, Year of Joining, and email id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.



TKR COLLEGE OF ENGINEERING & TECHNOLOGY
(Autonomous, Accredited by NAAC with „A Grade)
B. Tech INFORMATION TECHNOLOGY – R20



7. Write a PHP program to keep track of the number of visitors visiting the web page and to display this count of visitors, with proper headings.
8. Write a PHP program to display a digital clock which displays the current time of the server.
9. Write the PHP programs to do the following: a. Implement simple calculator operations. b. Find the transpose of a matrix. c. Multiplication of two matrices
10. Write a PHP program named states.py that declares a variable state with value "Mississippi Alabama Texas Massachusetts Kansas".
11. Write a PHP program to sort the student records which are stored in the database using selection sort.
12. A web application that lists all cookies stored in the browser on clicking "List Cookies" button. Add cookies if necessary.

REFERENCEBOOKS:

1. The Complete Reference PHP– Steven Holzner, Tata McGraw-Hill
2. Web Programming, building internet applications, Chris Bates 2nd edition, Wiley
Dream tech
3. Java Server Pages–Hans Bergsten, SPDO“ Reilly
4. Java Script, D. Flanagan, O“ Reilly, SPD.
5. Internet