Adikavi Nannaya University
Course Structure and Syllabus

II BTech IT II Semester

(From the admitted batch of 2017 – 2018 under CBCS Scheme)

Sub Code	Subject	Hrs/Week		Max Marks		Total Marks	Credits
		Theory	Lab	Internal	External		
BTINF401	COMPUTER ORGANIZATION	4	--	25	75	100	4
BTINF402	DESIGN AND ANALYSIS OF ALGORITHMS	4	--	25	75	100	4
BTINF403	DATABASE MANAGEMENT SYSTEMS	4	--	25	75	100	4
BTINF404	UNIX AND SHELL PROGRAMMING	4	--	25	75	100	4
BTINF405	SOFTWARE ENGINEERING	4	--	25	75	100	4
BTINF406	DISCRETE MATHEMATICAL STRUCTURES	4	--	25	75	100	4
BTINF407	UNIX AND SHELL PROGRAMMING LAB	--	3	50	50	100	2
BTINF408	DATABASE MANAGEMENT SYSTEMS LAB	--	3	50	50	100	2
TOTAL		**24**	**6**	**250**	**550**	**800**	**28**

BTINF401: COMPUTER ORGANIZATION

Theory	**: 4 Hrs/week**	**Credits**	**: 4**
Int Marks	**: 25**	**Ext Marks**	**: 75**

UNIT-I

Basic Computer Organization: Basic Computer Organization: Instruction Codes-Stored Program Organization, Indirect Address, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory Reference Instructions, Design of Accumulator Logic, General Register Organization, Stack Organization, Instruction Formats, Addressing Modes.

UNIT-II

Input-Output Organizations: Input-Output Organization: Input-Output Interface, Asynchronous Data Transfer—strobe control, hand shaking, asynchronous communication interface, Modes of Transfer, Direct Memory Access(DMA)-DMA Controller,DMA Transfer, Input-Output Processor (IOP).

UNIT-III

Memory Organizations: Memory Hierarchy, Main Memory, RAM, ROM Chips, Memory Address Map, Memory Connection to CPU, Associative Memory, Cache Memory, Miss and Hit Ratio, Access Time, Associative, Set Associative Mapping, Introduction to Virtual Memory.

UNIT-IV

8085 CPU: Introduction to Intel 8085 microprocessor Architecture-ALU,Timing and Control Unit,Register,Data&Address Bus, 8085 pin configuration, Instruction set of Intel 8085-Data Transfer Group, Arithmetic Group, Logical Group,Branch &Control Group, Addressing Modes of 8085,Assembly Language programs involving evaluation of Arithmetic Expressions.

TEXT BOOKS:

1. Computer System Architecture: Morris Mano.
2. Fundamentals of Microprocessor and Microcomputers: B.Ram-DhanpatRai Publications.

REFERENCE BOOKS:

1. Computer Organization and Architecture – William Stallings Sixth Edition, Pearson/PHI.
2. Structured Computer Organization and Design - Andrew S. Tanenbaum, 4th Edition PHI/Pearson.
3. Fundamentals or Computer Organization and Design –SivaraamaDandamudi Springer Int. Edition.
4. Computer Architecture a quantitive approach, Jhon L. Hennessy and David A. Patterson, Fourth Edition Elsevier.

BTINF402:DESIGN AND ANALYSIS OF ALGORITHMS

Theory	**: 4 Hrs/week**	**Credits**	**: 4**
Int Marks	**: 25**	**Ext Marks**	**:75**

UNIT-I

Introduction: Definition of Algorithm, Algorithm Specification, Performance Analysis.

Fundamentals of analysis of algorithms and efficiency: Analysis framework, Asymptotic Notations and Basic Efficiency classes, Mathematical Analysis of Non-recursive Algorithms, Mathematical Analysis of recursive Algorithms, Empirical Analysis of Algorithms, Algorithm Visualization.

Brute Force: Selection Sort, Bubble sort, Sequential Search, Brute-Force String Matching, Closest-Pair and Convex-Hull Problems by Brute Force, and Exhaustive Search.

UNIT-II

Divide-and-Conquer: Merge Sort, Quick sort, Binary Search, Binary Tree Traversals and Related Properties, Closest-Pair and Convex-Hull Problems by Divide and Conquer.

Decrease-and-Conquer: Insertion Sort, Depth-First Search and Breadth-First Search- Topological Sorting, Decrease-by-a-Constant-Factor Algorithms, Variable-Size-Decrease Algorithms.

Transform-and-Conquer: Presorting, Balanced Search Trees, Heaps and Heap sort, Horner’s Rule and Binary Exponentiation. Problem Reduction.

UNIT-III

Space and Time Tradeoffs: Sorting by Counting, Hashing, Input Enhancement in String Matching, B-Trees.**Dynamic Programming:** Warshall’s and Floyd’s Algorithm, Optimal Binary Search Trees, The Knapsack Problem and Memory Functions.

Greedy Technique: Prim’s Algorithm, Kruskal’s Algorithm, Dijkstra’s Algorithm, Huffman Trees

UNIT-IV

Limitations of Algorithm Power: Lower-Bound Arguments, Decision Trees, P, NP and NP-complete problems.

Coping with the Limitations of Algorithms Power: Backtracking: n-Queen Problem, Subset-Sum Problem, Branch-and-Bound: Knapsack Problem & TSP, Approximation Algorithms for NP-hard Problems.

Text Book:

[1] Introduction to Design & Analysis of Algorithms by Anany Levitin, Pearson Education, New Delhi.
[2] Fundamentals of Computer Algorithms E. Horowitz S. Sahani, S. Rajasekaran, Galgotia Publications.

Reference Books:

[1] Introduction to Algorithms by Thomas H. Corman, Charles E. Leiserson, Ron ald R. Rivest & Clifford Stein, Prentice Hall of India, New Delhi.
[2] The Design and Analysis of computer Algorithms, Aho, Hopcroft & Ullman, Pearson Education, New Delhi.

BTINF403: DATABASE MANAGEMENT SYSTEMS

Theory	**: 4 Hrs/week**	**Credits**	**: 4**
Int Marks	**: 25**	**Ext Marks**	**: 75**

UNIT I

Database System Introduction: Data, DBMS, Database Users, Advantages and Applications of DBMS

DBMS Concepts: Data Models, Schemas and Instances, Three-Schema Architecture and Data Independence, Database Languages and Interfaces, The Database System Environment, Centralized and Client/Server Architectures for DBMS, Classification of DBMS.

Data Modeling Using the Entity-Relationship Model: Using High-Level Conceptual Data Models for Database Design, Entity Types, Entity Sets Attributes and Keys, Relationships Types, Relationship Sets, Roles and Structural Constraints, Weak Entity types, ER diagrams, Naming Conventions and Design Issues, Enhanced Entity Relationship model.

UNIT II

Relational Data Model and Relational Database Constraints: Relational Model Concepts, Constraints and Relational Schemas, Update Operations and Dealing with Constraint Violations, Relational Database Design Using ER to Relational Mapping.

Relational Algebra and Relational Calculus: Unary Relational Operations, Relational Algebra Set Operations, Binary Relational Operations, Additional Relational Operations, Examples of Queries in Relational Algebra, Tuple and Domain Relational Calculus.

Schema Definition, Basic Constraints and Queries: SQL Data definition, Specifying Basic Constraints in SQL, Schema Change Statements in SQL, Insert,Delete,Update Queries in SQL, More Complex SQL queries,Views in SQL, Data base Stored Procedures.

UNIT III

Relational Database Design: Informal Design Guidelines for Relation Schema, Functional Dependencies, Normal Forms Based on Primary keys, General definitions of Second and Third Normal forms, BCNF, Properties of Relational Decomposition, Algorithm for Relational Database Design, Multi-Valued Dependencies and Fourth Normal form

Indexing Structures for files: types of single level ordered indexes, multilevel indexes, dynamic multilevel indexes using B Trees and B + TREES, Indexes on multiple keys.

UNIT IV

Transaction Processing: Introduction to Transaction Processing, Transaction and System Concepts, Desirable Characteristics of Transactions, Characterising Schedules Based On Recoverability and Serializability

Concurrency Control Techniques: Two Phase Locking, Timestamp Ordering, Validation Concurrency Control, Multiple Granularity Locking

Database Recovery Techniques: Recovery Concepts, Recovery Based On Deferred and Immediate Updates, Shadow Paging

TEXT BOOK:

1. Fundamentals of Database System,Elmasri, Navathe, Pearson Educaiton.

REFERENCES BOOKS:

1. Database Management Systems, Raghu Ramakrishnan,JohannesGehrke,McGraw-Hill.
2. Database Concepts, Abraham Silberschatz, Henry F Korth, S Sudarshan, McGraw-Hill

BTINF404 UNIX AND SHELL PROGRAMMING

Theory	**: 4 Hrs/week**	**Credits**	**: 4**
Int Marks	**: 25**	**Ext Marks**	**: 75**

Unit - I

Introduction to Unix:- Architecture of Unix, Features of Unix, Unix Commands: PATH, man, echo, printf, script, passwd, uname, who, date, stty, pwd, cd, mkdir, rmdir, ls, cp, mv, rm, cat, more, wc, lp, od, tar, gzip.

Unix Utilities: Introduction to unix file system, vi editor, file handling utilities, security by file permissions, process utilities, disk utilities, networking commands, unlink, du, df, mount, umount, find, unmask, ulimit, ps, w, finger, arp, ftp, telnet, rlogin.Text processing utilities and backup utilities, detailed commands to be covered are tail, head, sort, nl, uniq, grep, egrep, fgrep, cut, paste, join, tee, pg, comm, cmp, diff, tr, awk, cpio.

Unit - II

File Management: File Structures, System Calls for File Management: create, open, close, read, write, lseek, link, symlink, unlink, stat, fstat, lstat, chmod, chown, Directory API:opendir, readdir, closedir, mkdir, rmdir, umask.

Introduction to Shells: Unix Session, Standard Streams, Redirection, Pipes, Tee Command, Command Execution, Command- Line Editing, Quotes, Command Substitution, Job Control, Aliases, Variables, Predefined Variables, Options, Shell/Environment Customization. Filters: Filters and Pipes, Concatenating files, Display Beginning and End of files, Cut and Paste, Sorting, Translating Characters, Files with Duplicate Lines, Count Characters, Words or Lines, Comparing Files.

Unit - III

Grep: Operation, grep Family, Searching for File Content. Sed : Scripts, Operation, Addresses, commands, Applications, grep and sed. awk: Execution, Fields and Records, Scripts, Operations, Patterns, Actions, Associative Arrays, String. Functions, String Functions, Mathematical Functions, User Defined Functions, Using System commands, in awk, Applications, awk and grep, sed and awk.

Unit - IV

Interactive Korn Shell: Korn Shell Features, Two Special Files, Variables, Output, Input, Exit Status of a Command, eval Command, Environmental Variables, Options, Startup Scripts, Command History, Command Execution Process. Korn Shell Programming: Basic Script concepts, Expressions, Decisions: Making Selections, Repetition, special Parameters and Variables, changing Positional Parameters, Argument Validation, Debugging Scripts, Script Examples

Text Books

1. Unix and shell Programming Behrouz A. Forouzan, Richard F. Gilberg. Thomson.
2. Your Unix the ultimate guide, Sumitabha Das, TMH. 2nd Edition. 2007-2008 Page 34 of 95.

Reference Books

1. Unix for programmers and users, 3rd edition, Graham Glass, King Ables, Pearson Education
2. Unix programming environment, Kernighan and Pike, PHI. / Pearson Education
3. The Complete Reference Unix, Rosen, Host, Klee, Farber, Rosinski, Second Edition, TMH.

BTINF405 : SOFTWARE ENGINEERING

Theory	**: 4 Hrs/week**	**Credits**	**: 4**
Int Marks	**: 25**	**Ext Marks**	**: 75**

UNIT-I

Software Engineering Process Models: The Software Problem, Cost, Schedule, and Quality, Scale and Change, Software Processes: Process and Project, Component Software Processes, Software Development Process Models: Waterfall Model, Prototyping, Iterative Development, Rational Unified Process, Timeboxing Model, Extreme Programming and Agile Processes, Using Process Models in a Project, Project Management Process

UNIT-II

Software Requirements Analysis and Specification: Value of a Good SRS, Requirement Process, Requirements Specification, Desirable Characteristics of an SRS, Components of an SRS, Structure of a Requirements Document, Data Flow Diagrams for analysis.

Planning: Effort Estimation, Top-Down Estimation Approach, Bottom-Up Estimation Approach, Project Schedule and Staffing, Quality Planning, Risk Management Planning, Risk Management Concepts, Risk Assessment, Risk Control, A Practical Risk Management Planning Approach, Project Monitoring Plan, Measurements, Project Monitoring and Tracking, Detailed Scheduling.

Software Architecture: Role of Software Architecture, Architecture Views, Component and Connector View: Components, Connectors, Architecture Styles for C&C View, Pipe and Filter, Shared-Data Style, Client-Server Style

UNIT-III

Design: Design Concepts, Coupling, Cohesion, The Open-Closed Principle, Function-Oriented Design, Structure Charts, Structured Design Methodology, An Example

Coding: Programming Principles and Guidelines, Structured Programming, Information Hiding, Some Programming Practices, Coding Standards, Incrementally Developing Code,An Incremental Coding Process, Test-Driven Development, Pair Programming. Managing Evolving Code: Source Code Control and Build, Refactoring,

UNIT-IV

Unit Testing: Testing Procedural Units, Code Inspection, Planning, Group Review Meeting, Metrics, Size Measures, Complexity Metrics

Testing: Testing Concepts, Error, Fault, and Failure, Test Case, Test Suite, and Test Harness, Levels of Testing, Testing Process, Test Plan, Test Case Design, Test Case Execution, Black-Box Testing, Equivalence Class Partitioning, Boundary Value Analysis, Pairwise Testing, State-Based Testing, White-Box Testing, Control Flow-Based Criteria, Test Case Generation and Tool Support, Metrics, Coverage Analysis, Reliability, Defect Removal Efficiency.

TEXT BOOK:

1. A Concise Introduction to Software Engineering – Pankaj Jalote, Springer Publications.

REFERENCE BOOKS:

1. Software Engineering: A Pratitioner’s Approach – Roger S Pressman, Seventh Edition McGrawHill International Edition.
2. Software Engineering, Ian Sommerville, Ninth edition, Pearson education.
3. Fundamentals of Software Engineering. Rajib Mall. Third Edition.PHI.

BTINF406: DISCRETE MATHEMATICAL STRUCTURES

Theory	**: 4 Hrs/week**	**Credits**	**: 4**
Int Marks	**: 25**	**Ext Marks**	**: 75**

UNIT – I

Foundations: Sets, Relations and Functions, Methods of Proof and Problem Solving Strategies, Fundamentals of Logic, Logical Inferences, Methods of Proof of an implication, First order Logic & Other methods of proof, Rules of Inference for Quantified propositions, Mathematical Induction.

UNIT – II

Elementary Combinatorics, Basics of Counting, Combinations and Permutations, Enumeration of Combinations and Permutations, Enumerating Combinations and Permutations with repetitions, Enumerating Permutation with Constrained repetitions. Recurrence relations, Generating functions of sequences, Calculating Coefficients of Generating Functions.

UNIT – III

Recurrence relations, Solving recurrence relations by Substitution and generating functions.The methods of characteristic roots, solutions of inhomogeneous recurrence relations.Relations and digraphs, Special properties of binary relations, Equivalence relations.Operations on relation,OrderingRelations,Lattices and Enumerations.

UNIT – IV

Trees: Trees and their Properties,SpanningTrees,DirecedTrees,BinaryTress.

Graphs: Basic Concepts, Isomorphisms and Subgraphs, Planar Graphs, Euler’s Formula; Multigraphs and Euler Circuits, Hamiltonian Graphs, Chromatic Numbers, The Four Color Problem.

TEXT BOOK:

1. Toe L.Mott, Abraham Kandel& Theodore P.Baker, Discrete Mathematics for Computer Scientists & Mathematicians, PHI 2nd edition,2016.

REFERENCE BOOKS:

1. J.P. Trembly and R. Manohar- Discrete Mathematics for Computer Scientists & Mathematicians, PHI Ltd., New Delhi, 2nd Edition, 2008
2. Kenneth H Rosen-Discrete Mathematics & its Applications , TMH, 6ht Edition, 2009

BTINF407 UNIX AND SHELL PROGRAMMING LAB

Lab	**: 3 Hrs/week**	**Credits**	**: 2**
Int Marks	**: 50**	**Ext Marks**	**: 50**

Programs: 1

Getting Familiar with Linux: Enter these commands at the UNIX prompt, and try to interpret the output: echo, passwd, date, hostname, uname –a, who, who am i, last, finger, w , top, man, clear, history, **File Handling Utilities:** cat, more, cc, ls, touch, unlink, vi, **Directory Handling Utilities:** pwd, rmdir, mkdir, cd, **Operations common to File and Directory:** mv, cp, ln, rm, **Process Utilities:** pid, ps, kill, at, nice, who, w, ulimit, fg, bg
File Level Security: chmod, chown, chgrp, **Disk Utilities:** du, df, mount, umount,
Networking Utilities: ftp, rlogin, telnet, finger, arp, **Text Processing Utilities:** head, tail, nl, sort, uniq, cut, paste, tr, join, tee, comm, cmp, diff, **Backup Utilities:** tar, gzip, cpio

Programs: 2

Session-1

a) Log into the system
b) Use *vi* editor to create a file called **myfile.txt** which contains some text
c) Correct typing errors during creation.
Ans: *Use arrow keys to move within the file and do corrections if needed*
d) Save the file
Ans: *Press **Esc**:wq (enter into control mode, writes the file and quit vi)*
e) logout of the system
Ans: *use exit or logout command to quit form unix*

Session-2

a) Log into the system
b) Open the file created in session 1
Note: enter into insert mode by pressing Insert Button and do C,*D,E* bits
c) Add some text
d) Change some text
e) Delete some text
f) Save the Changes
g) Logout of the system

Programs: 3

a) Log into the system
b) Use the cat command to create a file containing the following data. Call it *mytable* use tabs to separate the fields. (delimeter)

1425	Ravi	15.65
4320	Ramu	26.27
6830	Sita	36.15
1450	Raju	21.86

c) Use the ***cat*** command to display the file, *mytable*.

d) Use the ***vi*** command to correct any errors in the file, *mytable*.

e) Use the ***sort*** command to sort the file *mytable* according to the first field. Call the sorted file my table (same name)

f) Print the file *mytable*

g) Use the cut and paste commands to swap fields 2 and 3 of *mytable*. Call it *mytable* (same name)

h) Print the new file, *mytable*

i) Logout of the system.

Programs: 4

1)

   a. Login to the system

   b. Use the appropriate command to determine your login shell

   c. Use the */etc/passwd* file to verify the result of step b.

   d. Use ***who*** command and redirect the result to a file called **myfile1**, Use the more command to see the contents of **myfile1**.

   e. Use the ***date*** and ***who*** commands in sequence (in one line) such that the output of ***date*** will display on the screen and the output of ***who*** will be redirected to a file called **myfile2**. Use the ***more*** command to check the contents of **myfile2**.

2)

   a. Write a sed command that deletes the first character in each line in a file.

   b. Write a sed command that deletes the character before the last character in each line in a file.

   c. Write a sed command that swaps the first and second words in each line in a file.

Programs: 5

a) Write a shell script that takes a command -line argument and reports on whether it is directory, a file, or something else.

b) Write a shell script that accepts one or more file name as arguments and converts all of them to uppercase, provided they exist in the current directory.

c) Write a shell script that determines the period for which a specified user is working on the system.

Programs: 6

a) Write a shell script that accepts a file name starting and ending line numbers as arguments and displays all the lines between the given line numbers.

b) Write a shell script that deletes all lines containing a specified word in one or more files supplied as arguments to it.

Programs: 7

a) Write a shell script that computes the gross salary of an employee according to the following rules:

b) Write a shell script that accepts two integers as its arguments and computers the value of first number raised to the power of the second number.

Programs: 8

a) Write an interactive file-handling shell program. Let it offer the user the choice of copying, removing, renaming, or linking files. Once the user has made a choice, have the program ask the user for the necessary information, such as the file name, new name and so on.

b) Write shell script that takes a login name as command – line argument and reports when that person logs in

c) Write a shell script which receives two file names as arguments. It should check whether the two file contents are same or not. If they are same then second file should be deleted.

Programs: 9

a) Write a shell script that displays a list of all the files in the current directory to which the user has read, write and execute permissions.

b) Develop an interactive script that ask for a word and a file name and then tells how many times that word occurred in the file.

c) Write a shell script to perform the following string operations:

   i. To extract a sub-string from a given string.

   ii. To find the length of a given string.

Programs: 10

Write a C program that takes one or more file or directory names ascommand line input and reports the following information on the file: (Note: Use stat/fstat system calls)

Programs: 11

Write C programs that simulate the following UNIX commands: (Use system calls)

a) mv b) cp

Programs: 12

Write a C program that simulates ***ls*** Command (Use system calls / directory API)

BTINF408: DATABASE MANAGEMENT SYSTEMS LAB

Lab	**: 3 Hrs/week**	**Credits**	**: 2**
Int Marks	**: 50**	**Ext Marks**	**: 50**

SQL

1) Simple queries to understand DDL, DML and DCL commands. Apply different constraints with Create and Alter commands. Exercise on delete cascade and on update cascade clauses.
2) Perform Insert, delete and update operations based on conditions.
3) Different ways of performing Join.
4) Set operations and sub queries.
5) Exercise Aggregate Functions using Group By, Having and Order By clauses
6) Creation and dropping of Views.
7) Demonstration of operators in SQL.
8) Queries to demonstrate Conversion Functions, String Functions, Date Functions and Numeric Functions.
9) Creation of Synonyms, Sequences and Indexes.
10) Granting and Revoking of privileges.

PL/SQL

1) Simple programs to understand PL/SQL
2) Write a PL/SQL program to demonstrate exception–handling
3) Demonstrate the working of Commit, Rollback and SavepointinPL/SQL block.
4) Develop a program that includes the features Nested If and Case expression.
5) Program development using While loops, For loops
6) Programs using Cursors
7) Programs development using creation of procedures and functions.
8) Working of Triggers.

REFERENCES:

1. Oracle: The Complete Reference by Oracle Press
2. Rick F Vander Lans, “Introduction to SQL”, Fourth Edition, Pearson Education, 2007.
3. Fundamentals of Database System,Elmasri, Navathe, Pearson Educaiton.
4. Oracle PL/SQL for Dummies, Michael Rosenblum, Paul Dorsey, Wiley Publications.
5. Microsoft Virtual Academy- mva.microsoft.com.