BENGALURU NORTH UNIVERSITY

TAMAKA, KOLAR- 563103

Curriculum/Syllabus
for
Undergraduate Programme

Bachelor of Computer Applications

Choice Based Credit System
As per State Education Policy - Karnataka

Faculty of Science

(With Effect from Academic Year 2024-25)

Version 1.0 (12th August, 2024)

BENGALURU NORTH UNIVERSITY

KOLAR - 563103

State Education Policy - 2024
(Semester Scheme)

Curriculum Structure for Bachelor of Computer Applications (BCA)

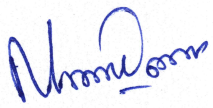
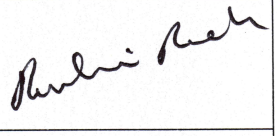
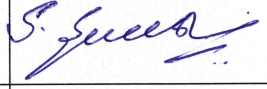
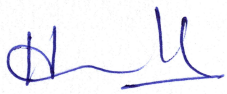
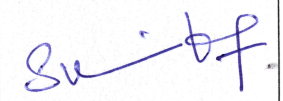
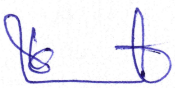
Syllabus for 1st and 2nd Semesters

With Effect From

Academic Year 2024 - 2025 and onwards

Curriculum Design/Syllabus Framing Committee

The following BOS members were present:

Sl.No.	Name and Address	Designation	Signature
1.	Dr. Murugan K, Associate Professor Department of Computer Science Government First Grade College K R Puram, Bengaluru-560036.	Chairperson	
2	Smt. Rashmi Rao K, Associate Professor Department of Computer Science Government First Grade College Hoskote, Bengaluru (Rural)	Member	
3	Dr. Sumanth S, Associate Professor Department of Computer Science Government College for Women Kolar – 563101	Member	
4	Dr. Rajendirakumar S, Associate Professor Department of Computer Science Government College for Women Kolar – 563101	Member	
5	Dr. Hamela K, Associate Professor Department of Computer Science Government First Grade College Malur, Kolar (Dist)	Member	
6	Mr. Manikandan S, Assistant Professor Department of Computer Science Government First Grade College K. R. Puram, Bengaluru	Member	
7	Dr. K. S. Manjunatha, Professor Department of Computer Science, Maharanis Science College for Women JLB Road, Mysore.	Member (External)	

Minutes:

1. The BoS members have approved the course structure and recommended (Theory and Lab).
2. The BoS members unanimously approved the proposed course structure for the newly introduced subjects.
3. The BoS members are accepted the detailed syllabi for the 1st and 2nd semester.

Regulations, Scheme of study and Examination for BCA Degree Course under Choice Based Credit System - Semester System (SEP Scheme)
(With effect from 2024 -2025)

R1**.** a) Title of the course: **Bachelor of Computer Applications**

b) Duration of the Course: Durations of the undergraduate programme shall extend SIX semesters (Three academic years) for the regular Bachelor Degree.

c) Scheme of study: there shall be six theory papers and two practical from first semester to five semesters.

d) There will be four theory, one practical and one project/Internship in sixth semester. The project work shall be carried out either independently or jointly (maximum of three students)

e) Medium of Instruction: The medium of instruction shall be English.

f) Scheme of Examination: At the end of each semester there is University Examination of three hours duration in each of the theory paper/practical.

R2. Each semester shall be of 90 working days from the date of commencement of the each Semester.

R3. Attendance: As per Bengaluru North University regulations in force for science degree courses.

R4. A Candidate is allowed to carry over all the previous uncleared (failed) theory papers/Practical to subsequent semesters as per Bengaluru North University regulations in force for science degree courses.

R5. The maximum period for completion of the course shall be as per Bengaluru North University regulations in force for science degree courses.

R6. Eligibility for admission:

a) A candidate who has passed the two years Pre-University Examination conducted by the Pre-University Education Board in Karnataka a minimum of 35% of marks.

b) A candidate who has passed Three years Diploma in Engineering of Government of Karnataka or any other examination considered as equivalent thereto shall be eligible for admission with minimum of 35% of marks in aggregate in all the semester /years.

c) Any student who has passed PUC –II Science, Arts or Commerce other than Karnataka securing with a minimum of 35% of marks.

R8. The total number of students to be admitted to the course shall be decided by the Bengaluru North University.

R9. Results: Results of candidate shall be declared and the classes awarded as per the procedure followed by the Bengaluru North University.

R10. Power to Remove Difficulties:

a) If may difficulty arises in giving effect to the provisions of these regulations, the Vice-Chancellor may be order make such provisions not inconsistent with the Act, Statutes, Ordinances or other Regulations, as appears to be necessary to expedient to remove the difficulty.

b) Every order made under this shall be subject to rectification by the appropriate University Authorities.

R11. The question paper pattern for theory paper has three sections. (80 Marks)

1. Section A includes 12 questions, students has to attend 10 questions. Each carries 2 Marks (10 * 2 = 20)
2. Section B includes 8 questions (question may contain sub questions), students has to attend 6 questions. each carries 5 Marks (6 * 5 = 30)
3. Section C includes 4 questions (question may contain sub questions), students has to attend 3 questions. each carries 10 Marks (3 * 10 = 30)

R12. Internal Assessment Scheme for Theory (4 Credits).

Assessment Criteria	Marks
Two test (Each carries 5 marks)	10
Assignment	5
Seminar	5
Total	20

R13. Internal Assessment Scheme for Theory (2 Credits).

Assessment Criteria	Marks
Test	5
Assignment	5
Total	10

R14. Internal Assessment Scheme for Practical (2 Credits).

Assessment Criteria	Marks
Test	5
Record	5
Total	10

R15. Evaluation Scheme for Practical Examination (SEE).

Assessment Criteria	Marks
Write up two programs (one from Section-A and Section-B)	20
Execution and output	15
Viva Voice based on Lab Activities	5
Total	40

Course Content for Bachelor of Computer Applications (BCA)

Curriculum Structure							
Program: BCA					**Subject: Computer Applications**		
Semester	**Course Code**	**Title of the Paper**	**Credits (L+P)**	**No.of Hours / Per Weak**	**Marks**		**Total Credits**
					SEE	**IA**	
I		Language-I	3+0	4	80	20	24
		English-I	3+0	4	80	20	
	CA1T1	Fundamentals of Computers	4+0	4	80	20	
	CA1T2	Programming in C	4+0	4	80	20	
	CA1T3	Computational Discrete Mathematics	4+0	4	80	20	
	CA1P1	Office Automation Lab	0+2	4	40	10	
	CA1P2	C Programming Lab	0+2	4	40	10	
		Constitutional Values-I	2+0	2	40	10	
II		Language-II	3+0	4	80	20	24
		English-II	3+0	4	80	20	
	CA2T1	Data Structures Using C	4+0	4	80	20	
	CA2T2	Statistical Methods using R Programming	4+0	4	80	20	
	CA2T3	Operating System Concepts	4+0	4	80	20	
	CA2P1	Data Structures Lab	0+2	4	40	10	
	CA2P2	R Programming Lab	0+2	4	40	10	
		Constitutional Values-II	2+0	2	40	10	

Semester: I
CA1T1: Fundamentals of Computers

Course Code: CA1T1	Course Title: Fundamentals of Computers
Course Credits: 04	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 20 Marks	Semester End Exam: 80 Marks

Course Outcomes (COs):

The student will be able to:

- Introduction to computers, classification of computers, anatomy of computer.
- Perform number conversions from one system to another system.
- Algorithm, Flowchart and Flowgorithm.
- Basics of Internet and E-mail, MS-Word, MS-Excel, MS-PowerPoint.

Course Content	Hours
Unit-I	15
Fundamentals of Computers: Introduction to Computers - Computer Definition, Characteristics of Computers, Evolution and History of Computers, Types of Computers, Basic Organisation of a Digital Computer. Computer Language and Software: Machine Language, Assembly Language, High Level Language, Assembler, Compiler, Interpreter. Software – System and Application Software. Algorithm, Flowchart and Pseudo code with Examples. Introduction to Flowgorithm, Flowgorithm Features.	
Unit-II	15
Input/Output Devices: Input Device – keyboard, mouse, scanner, MICR, OMR. Output Devices – VDU, Printers – Dot Matrix, line printers and page printers. Computer Memory: Memory Concept, Memory Cell, Memory Organisation, Semiconductor Memory – RAM, ROM, PROM, EPROM, Secondary Storage Devices – Magnetic Tape, Magnetic Disk-Floppy Disk, Hard Disk, Compact Disk. Logic Gates: The Inverter, The AND gate, The OR gate, The NAND gate, NOR gate, The Exclusive–OR gate and Exclusive-NOR gate.	
Unit-III	15
Number Systems – Different types, conversion from one number system to another; Computer Codes – BCD, Gray Code, ASCII and Unicode; Introduction of Internet and email: Features of Internet, Internet applications, web browser, search engine, e-mail, How to create e-mail, E-mail operations, E-mail-attaching a document,	
Unit-IV	15
MS Office: Introduction to MS Office, Components and Features. MS Word: Creating Letter, Table, Fonts, Page Layout Document, Formatting, Spell Check, Print Preview, Template, Color, Mail Merge, Auto Text, Inserting Picture, Word Art. MS Excel: Introduction to Excel, Sorting, Graphs, Scientific Functions. PowerPoint: Introduction to PowerPoint, Creation of Slides, Inserting Pictures, Preparing Slide Show with Animation.	

References:

1. Pradeep K. Sinha and Priti Sinha: Computer Fundamentals (Sixth Edition), BPB Publication
2. Floyd, Thomas L, "Digital Computer Fundamentals", 10 th Edition, University Book Stall,
3. Bartee, Thomas C, "Digital Computer Fundamentals", 6th Edition, TMH.
4. Introduction to Computer Science, ITL Education Solutions, Pearson Education

CA1T2: Programming in C

Course Code: CA1T2	Course Title: Programming in **C**
Course Credits: 04	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 20 Marks	Semester End Exam: 80 Marks

Course Outcomes (COs):

After completing this course satisfactorily, a student will be able to:

- Confidently operate Desktop Computers to carry out computational tasks.
- Understand working of Hardware and Software and the importance of operating systems.
- Understand programming languages, number systems, peripheral devices, networking, multimedia and internet concepts.
- Read, understand and trace the execution of programs written in C language.
- Write the C code for a given problem.
- Perform input and output operations using programs in C.
- Write programs that perform operations on arrays, strings, pointers and files.

Course Content	Hours
Unit-I	15
Overview of C Language: History of C, Character set, C tokens, Identifiers, Keywords, Data types, Variables, Constants, Symbolic Constants , Operators in C, Hierarchy of Operators, Expressions, Type Conversions and Library Functions. Managing Input and Output Operation: Formatted and Unformatted I/O Functions.	
Unit-II	15
Decision making, branching and looping: Decision Making Statements - if Statement, if–else statement, nested if statement, else–if ladder, switch statement, Looping - while, do-while, for loop, Nested loop, break, continue, and goto statements. Functions: Function Definition, prototyping, types of functions, passing arguments to functions, Nested Functions, Recursive functions.	
Unit-III	15
Arrays: Declaring and Initializing, One Dimensional Arrays, Two Dimensional Arrays, Multi Dimensional Arrays - Passing arrays to functions. Strings: Declaring and Initializing strings, Operations on strings, Arrays of strings, passing strings to functions. Storage Classes - Automatic, External, Static and Register Variables. Structures-Declaring and Initializing, Nested structure, Array of Structure, Passing Structures to functions, Unions, typedef, enum, Bit fields.	
Unit-IV	15
Pointers – Declarations, Pointer arithmetic, Pointers and functions, Call by value, Call by reference, Pointers and Arrays, Arrays of Pointers, Pointers and Structures. Meaning of static and dynamic memory allocation, Memory allocation functions. Files - File modes, File functions, and File operations, Text and Binary files, Command Line arguments. C Preprocessor directives, Macros – Definition, types of Macros, Creating and implementing user defined header files.	

References**:**

1. C: The Complete Reference, By Herbert Schildt.
2. C Programming Language, By Brain W. Kernighan
3. Kernighan & Ritchie: The C Programming Language (PHI)
4. P. K. Sinha & Priti Sinha: Computer Fundamentals (BPB)
5. E. Balaguruswamy: Programming in ANSI C (TMH)

CA1T3: Computational Discrete Mathematics

Course Code: CA1T3	Course Title: Computational Discrete Mathematics
Course Credits: 04	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 20 Marks	Semester End Exam: 80 Marks

Course Outcomes (COs):

After completing this course satisfactorily, a student will be able to:

- To understand the basic concepts of mathematical reasoning, set and functions.
- To understand various counting techniques and principle of inclusion and exclusion.
- Understand the concepts of various types of relations, partial ordering and equivalence relations.
- Apply the concepts of generating functions to solve the recurrence relations.
- Familiarize the fundamental concepts of graph theory and shortest path algorithm.

Course Content	Hours
Unit-I	15
The Foundations: Basic Concepts, Propositions, Truth Table, Connectives and Compound Propositions, Implication, Biconditional of Connectives, Converse, Inverse and Contra positive of an Implication, Tautology, Contradiction, Logical Equivalence, Applications of Propositional Logic, Propositional Equivalences. Basic Structures: Definition, Types of sets, Operation on Sets, Union, Intersection and Complements of Sets, Cartesian Product, Cardinality of Set. Determinants: Definition, Minors, Cofactors, Properties of Determinants. Matrices: Definition, Types of Matrices, Addition, Subtraction, Scalar Multiplication and Multiplication of Matrices, Adjoint, Inverse, Cramer's Rule.	
Unit-II	15
Counting: Basics of counting, Pigeonhole principle, Permutation and combination, Binomial Coefficient and Combination, Generating Permutation and Combination**.** Advanced Counting Techniques: Applications of Recurrence Relations, Solving Linear Recurrence Relations, Generating functions, Inclusion-Exclusion, Applications of Inclusion-exclusion.	
Unit-III	15
Induction and Recursion: Mathematical Induction, Strong Induction and Well-Ordering, Recursive Definitions and Structural Induction, Recursive Algorithms, Program Corrections. Relations and Functions: Properties of Relations, Equivalence Relation, Partial Order Relation Function: Domain and Range, Onto, Into and One to One Functions, Composite and Inverse Functions.	
Unit-IV	15
Graphs: Graphs and Graph models, Graph Terminology and Special Types of Graphs, Representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamilton Paths, Shortest-Path Problems, Planar Graphs, Graph Coloring.	

References:

1. Discrete Mathematics and Its Applications, Kenneth H. Rosen: Seventh Edition.
2. Discrete Mathematical Structure, Bernard Kolman, Robert C, Busby, Sharon Ross.
3. Graph Theory with Applications to Engg and Comp. Sci: Narsingh Deo-PHI.
4. Discrete and Combinatorial Mathematics Ralph P. Grimaldi, B. V. Ramatta, Pearson, Education, 5 Edition.
5. Discrete Mathematical Structures, Trembley and Manohar.

CA1P1: Office Automation Lab

Course Code: CA1P1	Course Title: Office Automation Lab
Course Credits: 02	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 10 Marks	Semester End Exam: 40 Marks

Section - A

1. Write a leave letter to the principal by using different alignments using MS-Word.

2. Create a bio-data using different alignments and use the page border using MS-Word.

.3. Create a time table of your class using MS-Word.

4. Create documents of your own and write the steps using MS-Word:

   a) Insert Pictures
   b) Insert Shapes

5. Create a documents using MS-Word.

   a) The word "MS-Word" as the watermark of the document.
   b) Set the background color of the document.
   c) Choose the indent tab.
   d) Change the space between paragraphs by adding space.

6. To prepare students mark sheet with the fields of Name, Register_Number, Mark1, Mark2, Mark3, Total, Average, Result and Class using MS-Excel.

7. To prepare employees payroll data with the fields of Sl.No. Name, Basic_pay, HRA, DA, PF, Gross_salary and Net_salary.

   a) Calculate HRA (10 % of Basic Pay), DA (25% of Basic Pay), DA (12% of Basic Pay).
   b) Calculate Gross_salary=Basic_pay+HRA+DA.
   c) Calcualte Net_salary=Gross_salary-PF

8. Prepare a bar chart and pie chart for analysis of five year results of your college using MS-Excel.

9. Create a line chart and bar chart using its data series using MS-Excel.

10. To prepare worksheet contains Name and Sales of 10 salesmen. Calculate commission as per the following:

Sales	Commission
First 30,000	5%
Next 40,000	10%
Excess	15%

Section - B

1. Create a power-point presentation with minimum 5 slides.

   a. The first slide must contain the topic of the presentation and name of the presentation.

   b. Second slide must contain at least 5 bullets, 5 numbers.

   c. The heading must be, font size:32, font-face: Arial Rounded MT Bold, font-color: blue.

   d. Last slide must contain "Thank you".

2. Create a power-point presentation with minimum 5 slides.

a. Use custom animation option to animate the text; the text must move left to right one line at a time.

b. Use proper transition for the slides.

3. Create a PowerPoint presentation with different animation format.

4. Create a PowerPoint presentation for company product by using different features.

5. Create a presentation about Computer Generations. The presentation should contain 5 slides, one for each generation. Apply transaction effect and slide advances in every two seconds automatically.

6. Using flowgorithm software, Execute different arithmetic tasks for sum, average, product, difference, quotient and remainder of given numbers.

7. Using flowgorithm software to calculate the area of shapes for square, rectangle, circle and triangle.

8. Using flowgorithm software,

a. Calculate the Fahrenheit to Celsius. F=(celsius*1.8)+32.

b. Calculate the Celsius to Fahrenheit. C=(Fahrenheit-32)/1.8.

9. Using flowgorithm software to check the given year is leap year or not.

10. Using flowgorithm software to find the largest of three integer numbers.

CA1P2: C Programming Lab

Course Code: CA1P2	Course Title: C Programming Lab
Course Credits: 02	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 10 Marks	Semester End Exam: 40 Marks

Section-A

1. Program to read radius of a circle and to find area and circumference.
2. Program to read three numbers and find the biggest of three.
3 Program to generate N primes.
4. Program to read a number, find the sum of the digits, reverse the number and check it for palindrome.
5. Program to read numbers from keyboard continuously till the user presses 999 and to find the sum of only positive numbers.
6. Program to read percentage of marks and to display appropriate message (Demonstration of else-if ladder).
7. Program to find the roots of quadratic equation (demonstration of switch case statement).
8. Program to read marks scored by N students and find the average of marks (Demonstration of single dimensional array).
9. Program to remove duplicate element in a single dimensional array.
10. Program to perform addition and subtraction of matrices

Section-B

1. Program to find the length of a string without using built in function

2. Program to demonstrate string functions.

3. Program to demonstrate pointers in C.

4. Program to check a number for prime by defining isprime( ) function.

5. Program to read, display and add two m x n matrices using functions.

6. Program to read, display and multiply two m x n matrices using functions.

7. Program to read a string and to find the number of alphabets, digits, vowels, consonants, spaces and special characters.

8. Program to reverse a string using pointer
9. Program to swap two numbers using pointers
10. Program to demonstrate student structure to read and display records of N students.

Semester: II

CA2T1: Data Structures using C

Course Code: CA2T1	Course Title: Data Structures using C
Course Credits: 04	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 20 Marks	Semester End Exam: 80 Marks

Course Outcomes (COs):

After completing this course satisfactorily, a student will be able to:

- Describe how arrays, records, linked structures, stacks, queues, trees, and graphs are represented in memory and used by algorithms
- Describe common applications for arrays, records, linked structures, stacks, queues, trees, and graphs
- Write programs that use arrays, records, linked structures, stacks, queues, trees, and graphs
- Demonstrate different methods for traversing trees.
- Describe the concept of recursion, give examples of its use.
- Discuss the computational efficiency of the principal algorithms for sorting and searching.

Course Content	Hours
Unit-I	15
Introduction and Overview: Definition, Elementary data organization, Data Structures, data structures operations, Abstract data types, Complexity of algorithms, asymptotic notations for complexity of algorithms. Arrays: Definition, Linear arrays, arrays as ADT, Representation of Linear Arrays in Memory, Traversing Linear arrays, Inserting and deleting elements, Multidimensional arrays, Representation of multidimensional Arrays and Sparse matrices.	
Unit-II	15
Sorting: Bubble sort, Insertion sort, Selection sort, Searching: Linear Search, Binary search. Linked list: Definition, Representation of Singly linked list in memory, Traversing a Singly linked list, Searching a Singly linked list, Memory allocation, Garbage collection, Insertion into a singly linked list, Deletion from a singly linked list; Doubly linked list, Header linked list, Circular linked list.	
Unit-III	15
Stacks – Definition, Array representation of stacks, Linked representation of stacks, Stack as ADT, Arithmetic Expressions: Polish Notation, Application of Stacks, Recursion, Towers of Hanoi, Implementation of recursive procedures by stack. Queues – Definition, Array representation of queue, Linked list representation of queues Types of queue: Simple queue, Circular queue, Double ended queue, Priority queue, Operations on Queues, Applications of queues	

Unit-IV	15
Tree – Definitions, Binary trees, Representing binary trees in memory, Traversal of binary tree; preorder, inorder and postorder traversal; Binary Search Trees, Searching, Inserting and Deleting in a Binary Search Tree. Graphs: Graph terminology, Sequential representation of Graphs: Adjacency matrix, Graph Traversals: Breadth First Search and Depth First Search.	

References:

1. Ellis Horowitz and Sartaj Sahni: Fundamentals of Data Structures
2. Tanenbaum: Data structures using C (Pearson Education)
3. Kamathane: Introduction to Data structures (Pearson Education)
4. Y. Kanitkar: Data Structures Using C (BPB)

CA2T2: Statistical Computing and R Programming

Course Code: CA2T2	Course Title: Statistical Computing and R Programming
Course Credits: 04	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 20 Marks	Semester End Exam: 80 Marks

Course Outcomes (COs):

After the successful completion of the course, the student will be able to:

- Explore fundamentals of statistical analysis in R environment.
- Describe key terminologies, concepts and techniques employed in Statistical Analysis.
- Define Calculate, Implement Probability and Probability Distributions to solve a wide variety of problems.
- Conduct and interpret a variety of Hypothesis Tests to aid Decision Making.
- Understand, Analyse, and Interpret Correlation Probability and Regression to analyse the underlying relationships between different variables.

Course Content	Hours
Unit-I	15
The Language: Introduction – Advantages of R over Other Programming Languages, R Studio: R script file, Handling Packages in R: Installing R Package, Syntax, Comments, Operators, R Keywords, R Data Types - numeric, Integer, logical, complex, character and raw, Variables, Input and Output statement, Data Structures – Strings, Vectors, Matrices, Arrays, Non-numeric Values, Lists and Data Frames, Special Values, Classes, and Coercion, Reading and Writing Files.	
Unit-II	15
Programming: Conditions and Loops - If statements - Stand-Alone Statement, Using If Else, Nesting and Stacking Statements, The Switch Function. Coding Loops - For Loops, While Loops, Repeat Loop, Other Control Flow Mechanisms - Declaring Break, Next and goto statement, R-Function: function definition, Built-in functions: Basic Math function - min(), max(), sum(), sqrt(),abs(),ceiling(),floor(), trunc(), round(), cos(), sin(), tan(), String function - grep(), nchar() , paste(), sprintf(), substr(), strsplit(), regex() gregexpr(), toupper(), tolower(), paste(), User Defined Function, Exception Handling, Progress and Timing, Visibility.	

Unit-III	15
Statistics and Probability: Elementary Statistics, Basic data visualisation, probability, common probability distributions: common probability mass functions - Bernoulli, Binomial, Poisson distributions, common probability density functions - Uniform, Normal, Student's t-distribution.	
Unit-IV	15
Statistical Testing and Modelling: Sampling distributions, hypothesis testing, Analysis of variance, Simple linear regression, multiple linear regressions. Advanced graphics: Basic Plotting, plot customization, plotting regions and margins, point and click coordinate interaction, customizing traditional R plots, specialized text and label notation, Defining colors and plotting in higher dimensions.	

References:

1 Tilman M. Davies, "The book of R: A first course in programming and statistics", San Francisco, 2016.
2 Vishwas R. Pawgi, "Statistical computing using R software", Nirali prakashan publisher, e1 edition, 2022.
3 https://www.geeksforgeeks.org/r-tutorial/
4. https://www.tutorialspoint.com/r/index.htm

CA2T3: Operating System Concepts

Course Code: CA2T3	Course Title: Operating System Concepts
Course Credits: 04	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 20 Marks	Semester End Exam: 80 Marks

Course Outcomes (COs):

At the end of the course, students will be able to:

- Explain the fundamentals of the operating system.
- Comprehend multithreaded programming, process management, process synchronization, memory management and storage management.
- Compare the performance of Scheduling Algorithms
- Identify the features of I/O and File handling methods.

Course Content	Hours
Unit-I	15
Introduction to Operating System: Definition, History and Examples of Operating System; Computer System organization; Types of Operating Systems; Functions of Operating System; Systems Calls; Operating System Structure. Process Management: Process Concept- Process Definition, Process State, Process Control Block, Threads; Process scheduling- Multiprogramming, Scheduling Queues, CPU Scheduling, Operations on Processes- Creation and Termination of Processes; Inter process communication (IPC)- Definition and Need for Inter process Communication, CPU Scheduling Criteria, Scheduling algorithm, Multiple Processor Scheduling, Real time Scheduling.	

Unit-II	15
Process Synchronization: Introduction; Race Condition; Critical Section Problem and Peterson's Solution; Synchronization Hardware, Semaphores; Classic Problems of Synchronization- Readers and Writers Problem, Dining Philosophers Problem; Monitors. Deadlocks: System Model; Deadlocks Characterization; Methods for Handling Deadlocks; Deadlock Prevention; Deadlock Avoidance; Deadlock Detection; and Recovery from Deadlock	
Unit-III	15
Memory Management: Logical and Physical Address Space; Swapping; Contiguous Allocation; Paging; Segmentation; Segmentation with Paging. Virtual Memory: Introduction to Virtual Memory; Demand Paging; Page Replacement; Page Replacement Algorithms; Allocation of frames, Thrashing.	
Unit-IV	15
File System: File Concepts- Attributes, Operations and Types of Files; File System; File Access methods; Directory Structure; Protection; File System Implementation- File System Structure, Allocation Methods, Free Space Management. Disk Structure & Scheduling methods, Disk management, Swap – Space management.	

References:

1. Operating System Concepts, Silberschatz' et al., 10thEdition, Wiley.
2. Operating System Concepts - Engineering Handbook, Ghosh PK.
3. Operating Systems - Internals and Design Principles, William Stallings, 9th Edition, Pearson.
4. Operating Systems – A Concept Based Approach, Dhamdhere, 3rd Edition, McGraw Hill Education India.
5. Modern Operating Systems, Andrew S Tanenbaum, 4th Edition, Pearson.

CA2P1: Data Structures Lab

Course Code: CA2P1	Course Title: Data Structures Lab
Course Credits: 02	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 10 Marks	Semester End Exam: 40 Marks

Section-A

1. Program to find GCD using recursive function
2. Program to display Pascal Triangle using binomial function
3. Program to generate n Fibonacci numbers using recursive function.
4. Program to implement Towers of Hanoi.
5. Program to implement dynamic array, find smallest and largest element of the array.
6. Program to create two files to store even and odd numbers.
7. Program to create a file to store student records.
8. Program to read the names of cities and arrange them alphabetically.
9. Program to sort the given list using selection sort technique.
10. Program to sort the given list using bubble sort technique.

Section-B

1. Program to sort the given list using insertion sort technique.
2. Program to sort the given list using quick sort technique.
3. Program to sort the given list using merge sort technique.
4. Program to search an element using linear search technique.
5. Program to search an element using recursive binary search technique.
6. Program to implement Stack.
7. Program to convert an infix expression to postfix.
8. Program to implement simple queue.
9. Program to implement linear linked list.
10. Program to display traversal of a tree.

CA2P2: R Programming Lab

Course Code: CA2P2	Course Title: R Programming Lab
Course Credits: 02	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 10 Marks	Semester End Exam: 40 Marks

Section-A

1. Write a R program that includes various data types in R.
2. Write a R program for different types of data structures in R
3. Write a R program that includes linear algebra operations on vectors and matrices.
4. Write a R program that includes various looping statements.
5. Write a R program for quick sort implementation, binary search tree.
6. Write a R program for calculating cumulative sums, products, minima, maxima and calculus.
7. Write a R program for finding stationary distribution of Markanov chains.
8. Write a R program for any visual representation of an object with creating graphs using graphic functions: Plot(),Hist(), Linechart(),Pie(),Boxplot(),Scatterplots().
9. Write a R program with any dataset containing data frame objects, indexing, and subsetting data frames, and employ manipulating and analyzing data.
10. Write a program to create any application of Linear Regression in a multivariate context for predictive purposes.

Section-B

1. Write a R program to take input from the user (name, age, address, city, state) and display the values. Also, print the version of the R installation.
2. Write a R program that includes different operators.
3. Write a R program to default values for arguments, returning complex objects.
4. Write a R program to create and store an array of 4×4 matrix and calculate its sum
5. Write a R program that includes various if statements.
6. Write a R program to calculate both simple interest and compound interest using functions.
7. Write a R program to implement various statistical functions in R.
8. Write a R program to find Correlation and Covariance.
9. Write a R program to import the data set and perform an ANOVA test.
10. Write a R program to define colors in various ways. (Use of named colors, RGB colors, hexadecimal color codes, and the colors()).

BENGALURU NORTH UNIVERSITY

KOLAR - 563103

State Education Policy - 2024
(Semester Scheme)

Curriculum Structure for Bachelor of Computer Applications (BCA)

Syllabus for 3rd and 4th Semesters

With Effect From

Academic Year 2024 - 2025 and onwards

Curriculum Design/Syllabus Framing Committee

The following BOS members were present:

Sl.No.	Name and Address	Designation	Signature
1.	Dr. Murugan K, Associate Professor Department of Computer Science Government First Grade College K R Puram, Bengaluru-560036.	Chairperson	
2	Smt. Rashmi Rao K, Associate Professor Department of Computer Science Government First Grade College Hoskote, Bengaluru (Rural)	Member	
3	Dr. Sumanth S, Associate Professor Department of Computer Science Government College for Women Kolar – 563101	Member	
4	Dr. Rajendirakumar S, Associate Professor Department of Computer Science Government College for Women Kolar – 563101	Member	
5	Dr. Hamela K, Associate Professor Department of Computer Science Government First Grade College Malur, Kolar (Dist)	Member	
6	Mr. Manikandan S, Associate Professor Department of Computer Science Government First Grade College K. R. Puram, Bengaluru	Member	
7	Dr. K. S. Manjunatha, Professor Department of Computer Science, Maharanis Science College for Women JLB Road, Mysore.	Member (External)	

Minutes:

1. The BoS members have approved the course structure and recommended (Theory and Lab).
2. The BoS members unanimously approved the proposed course structure for the newly introduced subjects.
3. The BoS members are accepted the detailed syllabi for the 3rd & 4th Semester

Regulations, Scheme of study and Examination for BCA Degree Course under Choice Based Credit System - Semester System (SEP Scheme)
(With effect from 2024 -2025)

R1**.** a) Title of the course: **Bachelor of Computer Applications**

b) Duration of the Course: Durations of the undergraduate programme shall extend SIX semesters (Three academic years) for the regular Bachelor Degree.

c) Scheme of study: there shall be six theory papers and two practical from first semester to five semesters.

d) There will be four theory, one practical and one project/Internship in sixth semester. The project work shall be carried out either independently or jointly (maximum of three students)

e) Medium of Instruction: The medium of instruction shall be English.

f) Scheme of Examination: At the end of each semester there is University Examination of three hours duration in each of the theory paper/practical.

R2. Each semester shall be of 90 working days from the date of commencement of the each Semester.

R3. Attendance: As per Bengaluru North University regulations in force for science degree courses.

R4. A Candidate is allowed to carry over all the previous uncleared (failed) theory papers/Practical to subsequent semesters as per Bengaluru North University regulations in force for science degree courses.

R5. The maximum period for completion of the course shall be as per Bengaluru North University regulations in force for science degree courses.

R6. Eligibility for admission:

a) A candidate who has passed the two years Pre-University Examination conducted by the Pre-University Education Board in Karnataka a minimum of 35% of marks.

b) A candidate who has passed Three years Diploma in Engineering of Government of Karnataka or any other examination considered as equivalent thereto shall be eligible for admission with minimum of 35% of marks in aggregate in all the semester /years.

c) Any student who has passed PUC –II Science, Arts or Commerce other than Karnataka securing with a minimum of 35% of marks.

R8. The total number of students to be admitted to the course shall be decided by the Bengaluru North University.

R9. Results: Results of candidate shall be declared and the classes awarded as per the procedure followed by the Bengaluru North University.

R10. Power to Remove Difficulties:

a) If may difficulty arises in giving effect to the provisions of these regulations, the Vice-Chancellor may be order make such provisions not inconsistent with the Act, Statutes, Ordinances or other Regulations, as appears to be necessary to expedient to remove the difficulty.

b) Every order made under this shall be subject to rectification by the appropriate University Authorities.

R11. The question paper pattern for theory paper has three sections. (80 Marks)

1. Section A includes 12 questions, students has to attend 10 questions. Each carries 2 Marks (10 * 2 = 20)
2. Section B includes 8 questions (question may contain sub questions), students has to attend 6 questions. each carries 5 Marks (6 * 5 = 30)
3. Section C includes 4 questions (question may contain sub questions), students has to attend 3 questions. each carries 10 Marks (3 * 10 = 30)

R12. Internal Assessment Scheme for Theory (4 Credits).

Assessment Criteria	Marks
Two test (Each carries 5 marks)	10
Assignment	5
Seminar	5
Total	20

R13. Internal Assessment Scheme for Theory (2 Credits).

Assessment Criteria	Marks
Test	5
Assignment	5
Total	10

R14. Internal Assessment Scheme for Practical (2 Credits).

Assessment Criteria	Marks
Test	5
Record	5
Total	10

R15. Evaluation Scheme for Practical Examination (SEE).

Assessment Criteria	Marks
Write up two programs (one from Section-A and Section-B)	20
Execution and output	15
Viva Voice based on Lab Activities	5
Total	40

Course Content for Bachelor of Computer Applications (BCA)

Curriculum Structure							
Program: BCA					**Subject: Computer Applications**		
Semester	**Course Code**	**Title of the Paper**	**Credits (L+P)**	**No. of Teaching Hours / Per Week**	**Marks**		**Total Credits**
					SEE	**IA**	
III		Language-III	3+0	4	80	20	26
		English-III	3+0	4	80	20	
	CA3T1	Object Oriented Concepts using JAVA	4+0	4	80	20	
	CA3T2	Database Management Systems	4+0	4	80	20	
	CA3T3	Design and Analysis of Algorithms	4+0	4	80	20	
	CA3P1	Java Programming Lab	0+2	4	40	10	
	CA3P2	DBMS Lab	0+2	4	40	10	
	SEC1	Shell Programming Lab	0+2	4	40	10	
	DSE1	Internet of Things / Cloud Computing	2+0	2	40	10	
IV		Language-IV	3+0	4	80	20	26
		English-IV	3+0	4	80	20	
	CA4T1	Python Programming	4+0	4	80	20	
	CA4T2	Artificial Intelligence & Applications	4+0	4	80	20	
	CA4T3	Computer Networks	4+0	4	80	20	
	CA4P1	Python Programming Lab	0+2	4	40	10	
	CA4P2	Artificial Intelligence Lab using Python	0+2	4	40	10	
	CMVII	Constitutional Moral Values-II	2+0	2	40	10	
	DSE2	Fundamentals of Data Science / Machine Learning	2+0	2	40	10	

Semester: III

CA3T1: Object Oriented Concepts using JAVA

Course Code: CA3T1	Course Title: Object Oriented Concepts using JAVA
Course Credits: 04	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 20 Marks	Semester End Exam: 80 Marks

Course Outcomes (COs): After the successful completion of the course, the student will be able to:

CO1	Understand the features of Java and the architecture of JVM.
CO2	Write, compile, and execute Java programs that may include basic data types and control flow constructs and how type casting is done
CO3	Identify classes, objects, members of a class and relationships among them needed for a specific problem and demonstrate the concepts of polymorphism and inheritance.
CO4	To demonstrate programs based on interfaces and threads, Explain the benefits of Java exceptional handling mechanism.
CO5	Write, compile, execute Java programs that include GUIs and event driven programming and also programs based on files

Course Content	Hours
Unit-I	15
Introduction to Java: Java History, Java Features, How Java differs from C, Java and Internet, Java and World Wide Web, Web Browsers, Hardware and Software Requirements, Java Support Systems, Java Environment. Overview of Java Language: Introduction, Structure of Java program, Implementing a Java Program, Java Virtual Machine, Command Line Arguments, Programming Style. Constants, Variables, and Data Types: Introduction, Tokens, Constants, Variables, Data Types, Declaration of Variables, Giving Values to Variables, Scope of Variables, Symbolic Constants, Type Conversions, Getting Values of Variables, Standard Default Values, Operators and Expressions: Introduction, Arithmetic Operators, Relational Operators, Logical Operators, Assignment Operators, Increment and Decrement Operators, Conditional Operators, Bitwise Operators, Special Operators, Arithmetic Expressions, Evaluation of Expressions, Precedence of Arithmetic Operators, Associativity, Mathematical Functions. Decision making and branching: if and switch statements, Decision making and looping: The while, do while and for loops, Jumps in Loops Labeled Loops.	
Unit-II	15
Classes, Objects and Methods: Introduction, Defining a Class, Adding Variables, Adding Methods, Creating Objects, Accessing Class Members, Constructors, Methods Overloading, Static Members, Nesting of Methods, Inheritance: Extending a Class, Types of inheritance, Overriding Methods, Final Variables and Methods, Finalizer methods, Abstract Methods and Classes, Visibility Control. Arrays, Strings and Vectors: Arrays, One-dimensional Arrays, Creating an Array, Two -Dimensional Arrays, Creating an Array, Two – dimensional Arrays, Strings, Vectors, Wrapper Classes.	
Unit-III	15
Interfaces: Introduction, Defining Interfaces, Extending Interfaces, Implementing Interfaces, Accessing Interface Variables. Packages: Putting Classes together: Introduction, Java API Packages, Using System Packages, Naming Conventions, Creating Packages, Accessing a	

Package, Using a Package, Adding a Class to a Package, Hiding Classes. Multithreaded Programming: Introduction, Creating Threads, Extending the Thread Class, Stopping and Blocking a thread, Life Cycle of a thread, Using Thread Methods, Thread Exceptions, Thread Priority, Synchronization, Implementing the 'Runnable' Interface. Managing Errors and Exception: Introduction, Types of Exception Handling Code, Multiple Catch Statements, Using Finally Statement, Throwing Our Own Exceptions, Using Exceptions for Debugging.	
Unit-IV	15
Applet Programming: Introduction, How Applets Differ from Applications, Preparing to Write Applets, Building Applet Code, Applet Life Cycle, Creating an Executable applet, Designing a Web Page, Applet Tag, Adding Applet to HTML File, running the Applet, More About HTML Tags, Displaying Numerical Values, Getting Input from the User. Graphics programming: Introduction, The Graphics Class, Lines and rectangles, circles, and Ellipses, Drawing Arcs, Drawing Polygons, Lines Graphs, Using Control Loops in Applets, Drawing Bar Charts. Managing Input/Output Files in Java: Introduction, Concept of Streams, Stream Classes, Byte Stream Classes, Character Stream Classes, Using Streams, Other Useful I/O Classes, Using the File Class, Input / Output Exceptions, Creation of Files, Reading / Writing Characters, Reading / Writing Bytes, Handling Primitive Data Types, Concatenating and Buffering Files, Interactive Input and output, Other Stream Classes.	

References:

1. Programming with Java, By E Balagurusamy – A Primer, Fourth Edition, Tata McGraw Hill Education Private Limited.
2. Core Java Volume I – Fundamentals, By Cay S. Horstmann, Prentice Hall
3. Object Oriented Programming with Java : Somashekara, M.T., Guru, D.S., Manjunatha, K.S
4. Java 2 - The Complete Reference – McGraw Hill publication.
5. Java - The Complete Reference, 7th Edition, By Herbert Schildt– McGraw Hill publication.

CA3T2: Database Management Systems

Course Code: CA3T3	Course Title: Database Management Systems
Course Credits: 04	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 20 Marks	Semester End Exam: 80 Marks

Course Outcomes (COs): After the successful completion of the course, the student will be able to:

CO1	Explain the various database concepts and the need for database systems.
CO2	Identify and define database objects, enforce integrity constraints on a database using DBMS.
CO3	Demonstrate a Data model and Schemas in RDBMS.
CO4	Identify entities and relationships and draw ER diagram for a given real-world problem.
CO5	Convert an ER diagram to a database schema and deduce it to the desired normal form.
CO6	Formulate queries in Relational Algebra, Structured Query Language, (SQL) for database manipulation.
CO7	Explain the transaction processing and concurrency control techniques

Course Content	Hours
Unit-I	15
Introduction to Databases: Database, Characteristics of database approach. Database users. Data models. Database schema. Database architecture. Data independence, interfaces, and classification of DBMS. Advantages of using DBMS. E-R Model: Entity-Relationship model: E-R Model Concepts: Entity, Entity types, Entity sets, Attributes, Types of attributes, key attribute, and domain of an attribute. Relationships between the entities. Relationship types, roles and structural constraints, degree and cardinality ratio of a relationship. Weak entity types, E -R diagram.	
Unit-II	15
Relational Data Model: Relational model concepts, characteristics of relations. Relational model constraints, Domain constrains, key constraints, primary and foreign key constraints, integrity constraints and null values. SQL: Introduction to SQL, Data types, Types of database languages, DDL, DML, DCL, TCL, Aggregate functions, grouping, nested sub queries, views.	
Unit-III	15
Relational Algebra: Basic Relational Algebra operations. Set theoretical operations on relations, Join Operation. Data Normalization: Anomalies in relational database design. Decomposition. Functional dependencies. Types of normal form, first, second and third normal form. Boyce-Codd normal form. PL/SQL, Introduction, Data types, Syntax of PL/SQL, Variables, Constants, Operators, If and Case statements, Looping, Exception handling, Cursors, Functions, Packages.	
Unit-IV	15
Query Processing Transaction Management: Introduction Transaction Processing. Single user & multiuser systems. Transactions: read & write operations. Need of concurrency control: The lost update problem, Dirty read problem. Types of failures. Transaction states. Desirable properties (ACID properties) of Transactions. Concurrency Control Techniques: Locks and Time stamp Ordering. Deadlock & Starvation.	

References:

1. Fundamentals of Database Systems, Ramez Elamassri, Shankant B. Navathe, 7th Edition, Pearson, 2015
2. An Introduction to Database Systems, Bipin Desai, Galgotia Publications, 2010.
3. Introduction to Database System, C J Date, Pearson, 1999.
4. Database Systems Concepts, Abraham Silberschatz, Henry Korth, S.Sudarshan, 6th Edition, McGraw Hill, 2010.
5. Database Management Systems, Raghu Rama Krishnan and Johannes Gehrke, 3rd Edition, McGraw Hill, 2002

CA3T3: Design and Analysis of Algorithms

Course Code: CA3T3	Course Title: Design and Analysis of Algorithms
Course Credits: 04	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 20 Marks	Semester End Exam: 80 Marks

Course Outcomes (COs): After the successful completion of the course, the student will be able to:

CO1	Understand the fundamental concepts of algorithms and their complexity, including time and space complexity, worst-case and average-case analysis, and Big-O notation.

CO2	Design algorithms for solving various types of problems, such as Sorting, Searching, Graph traversal, Decrease-and-Conquer, Divide-and-Conquer and Greedy Techniques.
CO3	Analyze and compare the time and space complexity of algorithms with other algorithmic techniques.
CO4	Evaluate the performance of Sorting, Searching, Graph traversal, Decrease-and-Conquer, Divide-and-Conquer and Greedy Techniques using empirical testing and benchmarking, and identify their limitations and potential improvements.
CO5	Apply various algorithm design to real-world problems and evaluate their effectiveness and efficiency in solving them.

Course Content	Hours
Unit-I	15
Introduction: What is an Algorithm? Fundamentals of Algorithmic problem solving, Fundamentals of the Analysis of Algorithm Efficiency, Analysis Framework, Measuring the input size, Units for measuring Running time, Orders of Growth, Worst-case, Bestcase and Average-case efficiencies. Asymptotic Notations and Basic Efficiency classes, Informal Introduction, O-notation, Ω-notation, θ-notation, mathematical analysis of non-recursive algorithms, mathematical analysis of recursive algorithms.	
Unit-II	15
Brute Force & Exhaustive Search: Introduction to Brute Force approach, Selection Sort and Bubble Sort, Sequential search, Exhaustive Search- Travelling Salesman Problem and Knapsack Problem, Depth First Search, Breadth First Search.	
Unit-III	15
Decrease-and-Conquer: Introduction, Insertion Sort, Topological Sorting Divide-and-Conquer: Introduction, Merge Sort, Quick Sort, Binary Search, Binary Tree traversals and related properties.	
Unit-IV	15
Greedy Technique: Introduction, Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Lower-Bound Arguments, Decision Trees, P Problems, NP Problems, NP Complete Problems, Challenges of Numerical Algorithms.	

References:

1. Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures in C, 2nd Ed, Universities Press, 2014.
2. Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.
3. Gilberg & Forouzan, Data Structures: A Pseudo-code approach with C, 2nd Ed, Cengage Learning,2014.
4. Reema Thareja, Data Structures using C, 3rd Ed, Oxford press, 2012.
5. Jean-Paul Tremblay & Paul G. Sorenson, An Introduction to Data Structures with Applications, 2nd Ed, McGraw Hill, 2013.

CA3P1: Java Programming Lab

Course Code: CA3P1	Course Title: Java Programming Lab
Course Credits: 02	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 10 Marks	Semester End Exam: 40 Marks

Section - A

1. Write a program to find factorial of list of number reading input as command line argument.

2. Write a program to display all prime numbers between two limits.

3. Write a program to sort list of elements in ascending and descending order and show the exception handling.

4. Write a program to implement all string operations.

5. Write a program to find area of geometrical figures using method.

6. Write a program to implement constructor overloading by passing different number of parameter of different types.

7. Write a program to create student report using applet, read the input using text boxes and display the o/p using buttons.

8. Write a program to calculate bonus for different departments using method overriding.

9. Write a program to implement thread, applets and graphics by implementing animation of ball moving.

10. Write a program to implement mouse events and keyboard events

Section - B

11. Write a java program to demonstrate the usage of switch case statement.

12. Program to print first N Fibonacci numbers.

13. Write a program to find product of two matrices.

14. Write a program to accept N string and sort in ascending order.

15. Program to demonstrate the usage of classes and objects.

16. Program to implement the concepts of multiple inheritance.

17. Write a program to demonstrate the concepts of multithreaded programming.

18. Write an applet program to find the sum of two integers using user input.

19. Write an applet program to draw human face.

20. Write an applet program to create Bar Charts.

CA3P2: DBMS Lab

Course Code: CA3P2	Course Title: DBMS Lab
Course Credits: 02	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 10 Marks	Semester End Exam: 40 Marks

Section-A

1. Execute a single line query and group functions.

2. Execute DDL Commands.

3. Execute DML Commands

4. Execute DCL and TCL Commands.

5. Implement the Nested Queries.

6. Implement Join operations in SQL

Section-B

7. Create views for a particular table
8. Implement Locks for a particular table
9. Write PL/SQL procedure for an application using exception handling.
10. Write PL/SQL procedure for an application using cursors.
11. Write a PL/SQL procedure for an application using functions
12. Write a PL/SQL procedure for an application using package

SEC1: Shell Programming Lab

Course Code: SEC1	Course Title: Shell Programming Lab
Course Credits: 02	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 10 Marks	Semester End Exam: 40 Marks

General Guidelines for Offering Skill Enhancement Course in BCA General, BCA (Data Science) and B.Sc. (Artificial Intelligence) Degree

1. SEC practical classes shall be conducted using the same operational framework as regular BCA/BCA(DS)/B.Sc. (AI) practical sessions with 0+0+2 credits for 0+0+4 practical hours per week requiring minimum of 60 practical Lab hours per subject per semester.

2. The guidelines issued by the Department of Collegiate Education (DCE) can be followed for making students batches for SEC practical for calculating corresponding workload for faculty.

- If the student strength is less than or equal to 19, a single faculty with 4 hours of workload can be assigned.
- If the student's strength is 20 to 30, two faculties can be assigned with 8 hours of practical workload.
- When student strength exceeds more than 30, practical classes may be conducted in parallel laboratories in the same semester.
- Internal Assessment (IA) scheme and Semester End Exam (SEE) to be followed as per regular practical examination.

List of Experiments

Section – A

1. To demonstrate the usage of any five internal commands.
2. To demonstrate the usage of any five external commands.
3. Write a shell script to calculate simple interest.
4. To print all prime numbers between m and (m<n).
5. Write a shell script to reverse a given number and check whether it is palindrome or not.

6. Write a shell script to find maximum and minimum of given set.
7. Write a shell script to generate and print the GCD and LCM of two integers.
8. Write a shell script to count lines, words and characters in its input. (do not use wc)
9. Write a shell script that displays the list of all files in the given directory.
10. Write a shell script to display files which has read, write and execute permissions

Section – B

11. Write a shell script to check given year is Leap year or not.
12. Write a shell script to make a basic calculator which performs addition, subtraction, Multiplication, division.
13. Write a shell script to count number of positive, negative and zero in a given set of numbers.
14. Write a shell script to input angles of a triangle and find out whether it is valid triangle or not.
15. Write a shell script to find factorial of given number.
16. Write a shell script to generate a multiplication table of the given number
17. Write a shell script to count number of digits and print sum of digit of any integer number.
18. Write a shell script to display Fibonacci series of n numbers.
19. Write a shell script to find reverse of a given string.
20. Write a shell script to check given string is palindrome or not.

DSE1: Internet of Things

Course Code: DSE1	Course Title: Internet of Things
Course Credits: 02	Teaching Hours/Week: 02
Total Contact Hours: 30	Exam Duration : 1 ½ Hours
Internal Assessment : 10 Marks	Semester End Exam: 40 Marks

Course Outcomes (COs): After the successful completion of the course, the student will be able to:

CO1	Understand the various concepts and architecture of IoT systems.
CO2	Understand various applications of IoT.
CO3	Use sensors and actuators for design of IoT.
CO4	Use various techniques of data storage and analytics in IoT.

Course Content	Hours
Unit-I	10
Introduction: Features of IoT, Characteristics of IoT, Architecture, Role of cloud in IoT, Applications of IoT: Home Automation, Smart Cities, Energy, Retail Management, Logistics, Agriculture, Health and Lifestyle, Industrial IoT, Legal challenges, IoT design Ethics, IoT in Environmental Protection.	
Unit-II	10
Sensing and Actuation: Definition of Sensor, Sensor features, Resolution, Classes, Different types of sensors, Actuator, Different types of Actuators, purpose of Sensors and Actuators in IoT	

Unit-III	10
IoT Application Development: Frame work for IoT Applications-Implementation of Device integration, Data acquisition and Integration, Device data storage on cloud/local server, Authentication, authorization of Devices.	

References:

1. Hakima Chaouchi, "The Internet of Things Connecting Objects to the Web" ISBN : 978-1- 84821-140-7, Wiley Publications
2. 2. Olivier Hersent, David Boswarthick, and Omar Elloumi, "The Internet of Things: Key Applications and Protocols", WileyPublications
3. Adrian Mcewen, Hakin Cassimally, "Designing The Internet of Things", First Edition, Wiley, 2014.
4. Keysight Technologies, "The Internet of Things: Enabling Technologies and Solutions for Design and Test", Application Note, 2016.
5. Daniel Minoli, "Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications", ISBN: 978-1-118-47347-4, Willy Publications

DSE1: Cloud Computing

Course Code: DSE1	Course Title: Cloud Computing
Course Credits: 02	Teaching Hours/Week: 02
Total Contact Hours: 30	Exam Duration : 1 ½ Hours
Internal Assessment : 10 Marks	Semester End Exam: 40 Marks

Course Outcomes (COs): After the successful completion of the course, the student will be able to:

CO1	To understand the principles of cloud architecture, models and infrastructure.
CO2	To gain knowledge about virtualization Infrastructure.
CO3	To explore and experiment with various Cloud deployment environments.
CO4	To learn about the security issues in the cloud environment.

Course Content	Hours
Unit-I	10
Cloud Computing Foundation : Introduction to Cloud Computing – Move toCloud Computing – Types of Cloud – Working of Cloud Computing.	
Unit-II	10
Cloud Computing Architecture : Cloud Computing Technology – Cloud Architecture – Cloud Modeling and Design - Virtualization : Foundation – Grid,Cloud and Virtualization – Virtualization and Cloud Computing.	
Unit-III	10
Data Storage and Cloud Computing : Data Storage – Cloud Storage – Cloud Storage from LANs to WANs – Cloud Computing Services : Cloud Services – Cloud Computing at Work – Cloud Security : Data Security – Cloud Applications.	

References:

1. Cloud Computing – A Practical Approach for Learning and Implementation, A.Srinivasan and J.Suresh, Pearson India Publications, 2014.
2. Cloud Computing: Principles and Paradigms, edited by Rajkumar Buyya, James Broberg, Andrzej, Wiley India Publications, 2011.
3. Derrick Rountree, Ileana Castrillo: "The Basics of Cloud Computing", Elsevier, 2014.
4. K Chandrashekaran: "Essentials of Cloud Computing", CRC Press, 2015

Semester: IV
CA4T1: Python Programming

Course Code: CA4T1	Course Title: Python Programming
Course Credits: 04	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 20 Marks	Semester End Exam: 80 Marks

Course Outcomes (COs): After the successful completion of the course, the student will be able to:

CO1	Explain the basic concepts of Python Programming.
CO2	Demonstrate proficiency in the handling of loops and creation of functions.
CO3	Identify the methods to create and manipulate lists, tuples and dictionaries.
CO4	Discover the commonly used operations involving file handling.
CO5	Interpret the concepts of Object-Oriented Programming as used in Python.
CO6	Develop the emerging applications of relevant fields using Python

Course Content	Hours
Unit-I	15
Introduction to Python: Features of Python - How to Run Python - Identifiers - Reserved Keywords - Variables - Comments in Python - Indentation in Python - Multi-Line Statements - Multiple Statement Group (Suite) - Quotes in Python - Input, Output and Import Functions - Operators. Data Types and Operations: Numbers – Strings – List – Tuple – Set – Dictionary – Data type conversion. Python Control Flow: Types of Control Flow; Control Flow Statements- if, else, elif, while loop, break, continue statements, for loop Statement; range () and exit () functions.	
Unit-II	15
Exception Handling: Types of Errors; Exceptions; Exception Handling using try, except and finally. Python Functions: Types of Functions; Function Definition- Syntax, Function Calling, Passing Parameters/arguments, the return statement; Default Parameters; Command line Arguments; Key Word Arguments; Recursive Functions; Scope and Lifetime of Variables in Functions. Strings: Creating and Storing Strings; Accessing Sting Characters; the str() function; Operations on Strings- Concatenation, Comparison, Slicing and Joining, Traversing; Format Specifiers; Escape Sequences; Raw and Unicode Strings; Python String Methods.	
Unit-III	15
File Handling: File Types; Operations on Files– Create, Open, Read, Write, Close Files; File Names and Paths; Format Operator. Object Oriented Programming: Classes and Objects; Creating Classes and Objects; Constructor Method; Classes with Multiple Objects; Objects as Arguments; Objects as Return Values; Inheritance- Single and Multiple Inheritance, Multilevel and Multipath Inheritance; Encapsulation- Definition, Private Instance Variables; Polymorphism- Definition, Operator Overloading.	
Unit-IV	15
GU Interface: The tkinter Module; Window and Widgets; Layout Management- pack, grid and place. Python SQLite: The SQLite3 module; SQLite Methods- connect, cursor, execute, close; Connect to Database; Create Table; Operations on Tables- Insert, Select, Update. Delete and Drop Records. Data Analysis: NumPy- Introduction to NumPy, Array Creation	

using NumPy, Operations on Arrays; Pandas- Introduction to Pandas, Series and Data Frames, Creating Data Frames from Excel Sheet and .csv file, Dictionary and Tuples. Operations on Data Frames. Data Visualisation: Introduction to Data Visualisation; Matplotlib Library; Different Types of Charts using Pyplot- Line chart, Bar chart and Histogram and Pie chart.

References:

1. Think Python How to Think Like a Computer Scientist, Allen Downey et al., 2nd Edition, Green Tea Press.
2. Introduction to Python Programming, Gowrishankar S et al., CRC Press, 2019.
3. Python Data Analytics: Data Analysis and Science Using Pandas, matplotlib, and the Python Programming Language, Fabio Nelli, Apress®, 2015
4. Advance Core Python Programming, MeenuKohli, BPB Publications, 2021.
5. Core PYTHON Applications Programming, Wesley J. Chun, 3rd Edition, Prentice Hall, 2012.
6. Automate the Boring Stuff, Al Sweigart, No Starch Press, Inc, 2015.

CA4T2: Artificial Intelligence & Applications

Course Code: CA4T2	Course Title: Artificial Intelligence & Applications
Course Credits: 04	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 20 Marks	Semester End Exam: 80 Marks

Course Outcomes (COs): After the successful completion of the course, the student will be able to:

CO1	Gain a historical perspective of AI and its foundations
CO2	Become familiar with basic principles of AI toward problem solving.
CO3	Get to know approaches of inference, perception, knowledge representation, and learning.
CO4	Understand the concepts of learning and neural networks.

Course Content	Hours
Unit-I	15
Introduction- What is Artificial Intelligence, Foundations of AI, History, AI - Past, Present and Future. Intelligent Agents- Environments- Specifying the task environment, Properties of task environments, Agent based programs-Structure of Agents , Types of agents- Simple reflex agents, Model-based reflex agents, Goal-based agents; and Utility-based agents.	
Unit-II	15
Problem Solving by Searching-Problem-Solving Agents, Well-defined problems and solutions, examples Problems, Searching for Solutions, Uninformed Search Strategies-Breadth-first search, Uniform-cost search, Depth-first search, Depth-limited search, Iterative deepening depth-first search, Bidirectional search, Greedy best-first search, A* Search, AO* search Informed (Heuristic) Search Strategies, Heuristic Functions	
Unit-III	15
Knowledge Representation - Knowledge-Based Agents, The Wumpus World , Logic, Propositional Logic, Propositional Theorem Proving, Effective Propositional Model Checking, Agents Based on Propositional Logic, First-Order Logic-Syntax and Semantics of First-Order Logic, Using First-Order Logic, Unification and Lifting Forward Chaining, Backward Chaining	
Unit-IV	15
Learning– Forms of Learning, Supervised Learning, Machine Learning - Decision Trees,	

Regression and Classification with Linear Models, Artificial Neural Networks, Support Vector Machines

References:

1. Saroj Kaushik, Artificial Intelligence, Cengage learning, 2014.
2. Elaine Rich, Kevin Knight, Artificial Intelligence, Tata McGraw Hill
3. Nils J. Nilsson, Principles of Artificial Intelligence, Elsevier, 1980
4. StaurtRussel, Peter Norvig, Artificial Intelligence: A Modern Approach, Pearson Education, 3rd Edition, 2009
5. George F Lugar, Artificial Intelligence Structure and strategies for complex, Pearson Education, 5th Edition, 2011

CA4T3: Computer Networks

Course Code: CA4T3	Course Title: Computer Networks
Course Credits: 04	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 20 Marks	Semester End Exam: 80 Marks

Course Outcomes (COs): After the successful completion of the course, the student will be able to:

CO1	Define various data communication components in networking. Describe networking with reference to different types of models and topologies.
CO2	Explain various Data Communications media. Describe the physical layer functions and components
CO3	Describe various Data link Layer Protocols.
CO4	Identify the different types of network devices and their functions within a network.
CO5	Analyze and Interpret various Data Kink Layer and Transport Layer protocols.
CO6	Explain different application layer protocols

Course Content	Hours
Unit-I	15
Introduction: Computer Network, Uses of Computer Networks, Network hardware, Network software, OSI Reference Models, TCP/IP Reference models. Physical Layer: Guided Transmission media, Wireless transmission, Communication Satellites, Digital Modulation and Multiplexing.	
Unit-II	15
Data link layer: Design issues: framing, Error detection and correction. Elementary data link protocols: simplex protocol, A simplex stop and wait protocol for an error-free channel, A simplex stop and wait protocol for noisy channel. Sliding Window protocols: A one-bit sliding window protocol, A protocol using Go-Back-N, A protocol using Selective Repeat. Medium Access sub layer: ALOHA, CSMA, CSMA/CD, CSMA/CA, Wireless LANs, Data link layer switching.	
Unit-III	15
Network Layer: Design issues, Routing algorithms: shortest path routing, Flooding, Hierarchical routing, Broadcast, Multicast, distance vector routing, Congestion Control Algorithms, Quality of Service, Internetworking, The Network layer in the internet.	
Unit-IV	15
Transport Layer: Transport Services, Elements of Transport protocols, Connection management, TCP and UDP protocols. Application Layer: Domain name system, SNMP, Electronic Mail; the World Wide Web, HTTP, Streaming audio and video.	

References:
1. Computer Networks -- Andrew S Tanenbaum, David. j. Wetherall, 5th Edition. Pearson Education/PHI.
2. An Engineering Approach to Computer Networks-S. Keshav, 2nd Edition, Pearson Education
3. Data Communications and Networking – Behrouz A. Forouzan. Third Edition TMH.
4. Larry L. Peterson and Bruce S. Davie, "Computer Networks A System Approach", 5th Edition, MKP, 2012

CA4P1: Python Programming Lab

Course Code: CA4P1	Course Title: Python Programming Lab
Course Credits: 02	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 10 Marks	Semester End Exam: 40 Marks

Section-A

1. Write simple Python program to display message on screen
2. Write simple Python program using operators:
   a) Arithmetic Operators b) Logical Operators c) Bitwise Operators
3. Write a Python program to create a calculator program.
4. Write simple Python program to demonstrate use of conditional statements:
   a) if statement b) if ... else statement c) Nested if statement
5. Write python program to demonstrate use of looping statements:
   a) while loop b) for loop c) Nested loops
6. Write python program to perform following operations on Lists:
   a) Create list b) Access list
   c) Update list (Add item, Remove item) d) Delete list
7. Write python program to perform following operations on Tuples:
   a) Create Tuple b) Access Tuple
   c) Update Tuple d) Delete Tuple
8. Write python program to perform following operations on Tuples:
   a) Create Set b) Access Set elements
   c) Update Set d) Delete Set
9. Write python program to perform following operations on Dictionaries:
   a) Create Dictionary b) Access Dictionary elements
   c) Update Dictionary d) Delete Set
10. Write Python program to demonstrate math built- in functions

Section-B

11. Write Python program to demonstrate string built — in functions _
12. Develop user defined Python function for given problem:
    a) Function with minimum 2 arguments b) Function returning values
13. Write a program in Python to demonstrate following operations:
    a) Method overloading b) Method overriding

14. Write a program in Python to demonstrate following operations:
    a) Simple inheritance b) Multiple inheritance
15. Write a program in Python to handle user defined exception.
16. Create a GUI using Tkinter module.
17. Drawing Line chart and Bar chart using Matplotlib.
18. Drawing Histogram and Pie chart using Matplotlib.
19. Create Array using NumPy and Perform Operations on Array.
20. Create Data Frame from Excel sheet using Pandas and Perform Operations on Data Frames.

CA4P2: Artificial Intelligence Lab using Python

Course Code: CA4P2	Course Title: Artificial Intelligence Lab using Python
Course Credits: 02	Teaching Hours/Week: 04
Total Contact Hours: 60	Exam Duration : 3 Hours
Internal Assessment : 10 Marks	Semester End Exam: 40 Marks

Section – A

1. Demonstrate the various python libraries used for machine learning.
2. Write a program to implement Breadth First Search.
3. Write a program to implement Depth First Search.
4. Write a program to implement A* Search algorithm.
5. Write a program to implement AO* Search algorithm.
6. Write a Program to implement Tic-Tac-Toe game.
7. Write a Program to implement 8-Puzzle problem.
8. Write a Program to implement Water-Jug problem.

Section – B

9. Write a Program to implement Travelling Salesman Problem.
10. Write a Program to implement Monkey Banana Problem.
11. Write a Program to implement Alpha-Beta Pruning.
12. Write a Program to implement 8-Queens Problem.
13. Write a Program to implement Missionaries-Cannibals Problems
14. Write a Program to implement Hill Climbing Algorithm.
15. Write a Program to implement Forward Chaining algorithm.
16. Write a Program to implement FOPL related problems.

DSE2: Fundamentals of Data Science

Course Code: DSE2	Course Title: Fundamentals of Data Science
Course Credits: 02	Teaching Hours/Week: 02
Total Contact Hours: 30	Exam Duration : 1 ½ Hours
Internal Assessment : 10 Marks	Semester End Exam: 40 Marks

Course Outcomes (COs): After the successful completion of the course, the student will be able to:

CO1	Understand the concepts of data and preprocessing of data.
CO2	Know simple pattern recognition methods
CO3	Understand the basic concepts of Clustering and Classification
CO4	Know the recent trends in Data Science.

Course Content	Hours
Unit-I	10
Data Mining: Introduction, Data Mining Definitions, Knowledge Discovery in Databases Vs Data Mining, DBMS Vs Data Mining, DM techniques, Problems, Issues and Challenges in DM, DM applications.	
Unit-II	10
Data Warehouse: Introduction, Definition, Multidimensional Data Model, Data Cleaning, Data Integration and transformation, Data reduction, Discretization.	
Unit-III	10
Mining Frequent Patterns: Basic Concept – Frequent Item Set Mining Methods -Apriori and Frequent Pattern Growth algorithms -Mining Association Rules. Mining Frequent Patterns:	

Basic Concept – Frequent Item Set Mining Methods - Apriori and Frequent Pattern Growth algorithms -Mining Association Rules.

References:

1. Jiawei Han and Micheline Kambar – "Data Mining Concepts and Techniques" Second Edition Elsevier Publications.
2. Arun K Pujari –"Data Mining Techniques" 4th Edition, Universities Press.
3. Pang-Ning Tan, Michael Steinbach, Vipin Kumar: Introduction to Data Mining, Pearson Education, 2012.
4. K.P.Soman, Shyam Diwakar, V.Ajay: Insight into Data Mining – Theory and Practice, PHI.
5. Pang-Ning Tan, Michael Steinbach, Vipin Kumar - "Introduction to Data Mining", Pearson Education.

DSE2: Machine Learning

Course Code: DSE2	Course Title: Machine Learning
Course Credits: 02	Teaching Hours/Week: 02
Total Contact Hours: 30	Exam Duration : 1.30 Hours
Internal Assessment : 10 Marks	Semester End Exam: 40 Marks

Course Outcomes (COs): After the successful completion of the course, the student will be able to:

CO1	Understand the basic theory underlying machine learning.
CO2	Understand the features of machine learning to apply on real world problems
CO3	Characterize the machine learning algorithms as supervised learning and unsupervised learning
CO4	Learn the concepts in Bayesian analysis from probability models and methods

Course Content	Hours
Unit-I	10
Introduction to Machine Learning: Introduction, Perspectives & Issues in ML, designing learning systems, Concepts of hypotheses, Version space, inductive bias, Performance metrics-accuracy, precision, recall, sensitivity, specificity, AUC, RoC.	
Unit-II	10
Supervised Learning: Decision Trees Learning: Basic algorithm (ID3), Issues in Decision Tree Learning—Overfitting, Solutions to overfitting. Instance-based learning: k-nearest neighbour learning. Support Vector Machines: Introduction, Handling data that are linearly separable. Artificial Neural networks: Introduction, Perceptrons, Multi-layer networks and back-propagation.	
Unit-III	10
Probabilistic and Stochastic Models: Bayesian Learning—Bayes theorem, Concept learning, Maximum likelihood, Bayes optimal classifier, Naive Bayes classifier. Expectation Maximization and Gaussian Mixture Models, Hidden MarKov models.	

References:

1. Ethem Alpaydin, "Introduction to Machine Learning", MIT Press, Prentice Hall of India, Third Edition 2014
2. Jiawci Han and Micheline Kambers and JianPei,"Data Mining—Concepts and Techniques", 3rd edition, Morgan Kaufman Pub
3. Charu C.Aggarwal, "Data Classification Algorithms and Applications", CRC Press, 2014.
4. CharuC.Aggarwal, "Data Clustering Algorithms and Applications", CRC Press, 2014.
5. "Machine Learning", Tom Mitchell, McGraw Hill Education(India),2013.