

Department of Informatics and Computational Sciences
Bachelor of Computer Application Programme (BCA) 2023-24
Minimum Total and Course Category wise Credit Requirements

S.No.	Broad Category of Courses	Minimum Credit Requirement
1	Discipline Centric Core Course(DCC)	72
2	Discipline Specific Elective Course(DSE)	18
3	Ability Enhancement Compulsory Course(AECC)	4
4	Skill Enhancement Course(SEC)	6
5	Dissertation, Project, Report of Field Study (DPR)	20
	Total	120

Level	Semester	Course Type	Course Code	Course Title	Delivery Type Per Week			Total Hours	Credit	Total Credits	Internal Marks	External Marks	Max Marks	Remarks
					L	T	P							
					L	T	P							
I	DCC	BCA5000T	IT and PC Package	L	T	-	60	4	6	20	80	100		
		BCA5000P	IT and PC Package Lab	-	-	P	60	2		20	80	100		
		BCA5001T	C Programing	L	T	-	60	4	6	20	80	100		
		BCA5001P	C Programming Lab.	-	-	P	60	2		20	80	100		
		BCA5002T	Data Base Management System	L	T	-	60	4	6	20	80	100		
		BCA5002P	Data Base Management System Lab	-	-	P	60	2		20	80	100		

5		AECC		AECC-I	L	-	-	30	2	2	20	80	100		
	II	DCC	BCA5003T	Object oriented programming g using C++	L	T	-	60	4	6	20	80	100		
			BCA5003P	Object oriented Programming Using C++ Lab.	-	-	P	60	2		20	80	100		
			BCA5004T	Data Structure	L	T	-	60	4		20	80	100		
				BCA5004P	Data StructureLab	-	-	P	60	2	6	20	80	100	
				BCA5005T	Computer Organization and Architecture(COA)	L	T	-	60	4	6	20	80	100	
				BCA5005P	Computer Organization and Architecture(COA) Lab	-	-	P	60	2		20	80	100	
		AECC		AECC-II	L	-	-	30	2	2	20	80	100		
Exit with Certificate in Computer Application (After 4 more exit credits in SEC)															
6	III	DCC	BCA6006T	Java Programming	L	T	-	60	4	6	20	80	100		
		DCC	BCA6006P	Java programming Lab	-	-	P	30	2		20	80	100		
		DCC	BCA6007T	Operating System	L	T	-	60	4		20	80	100		
		DCC	BCA6007P	Operating System Lab.	-	-	P	30	2		6	20	80	100	
		DCC	BCA6008T	Accounting AndFinancial Management	L	T	-	60	4		20	80	100		
		DCC	BCA6008P	Accounting And Financial Management Lab	-	-	P	30	2		6	20	80	100	
	SEC		Communicative English	L	-	-	30	2	2	20	80	100			
6	IV	DCC	BCA6009T	System Analysis and Design	L	T	-	60	4	6	20	80	100		
		DCC	BCA6009P	System Analysis and Design Lab	-	-	P	60	2		20	80	100		

		DCC	BCA6010T	C # with Dot NetFramework	L	T	-	60	4		20	80	100				
		DCC	BCA6010P	C # with Dot NetFramework Lab	-	-	P	60	2	6	20	80	100				
		DCC	BCA6011T	Computer Communicationand Networks	L	T	-	60	4		20	80	100				
		DCC	BCA6011P	Computer Communicationand Networks Lab	-	-	P	60	2	6	20	80	100				
		SEC	SEI633XT	330T) Basic Mathematics 331T) Fundamental of Web Development 332T) Business Organization and Management	L	-	-	30	2	2	20	80	100				
Exit with Diploma in Computer Application (After 4 more exit credits in SEC)																	
7	V	DSE	BCA710XT	DSE-A1	L	T	-	60	4		20	80	100				
		DSE	BCA710XP		-	-	P	60	2		20	80	100				
		DSE	BCA710XT	DSE-B1	L	T	-	60	4		20	80	100				
		DSE	BCA710XP		-	-	P	60	2		20	80	100				
		DSE	BCA71XXT	DSE-C1	L	T	-	60	4		20	80	100				
		DSE	BCA71XXP		-	-	P	60	2		20	80	100				
		SEC	SEI733XP	SEC-Group	L	-	P	30	2	2	20	80	100				
		VI	DSE	BCA711XP	DSE-D1	-	-	P	540	18	18	20	80	100			
		SEC	SEI7339S	Report Writing	L	T	-	30	2	2	20	80	100				
Total												120	120	720	2880	3600	

Discipline Specific Elective Course (DSE)			
Code	Paper	Credits	Total Hours
DSE Group–A1			
BCA7100T	Introduction to Data Mining	4	60
BCA7100P	Introduction to Data Mining – LAB	2	60
BCA7101T	Python Programming	4	60
BCA7101P	Python Programming – LAB	2	60
BCA7102T	Cloud Computing	4	60
BCA7102P	Cloud Computing – LAB	2	60
BCA7103T	Discrete Mathematical Structure	4	60
BCA7103P	Discrete Mathematical Structure - LAB	2	60
DSE Group–B1			
BCA7104T	Internet of Things	4	60
BCA7104P	Internet of Things – LAB	2	60
BCA7105T	Desktop Publishing	4	60
BCA7105P	Desktop Publishing – LAB	2	60
BCA7106T	Mobile Application Development	4	60
BCA7106P	Mobile Application Development – LAB	2	60
BCA7107T	Advance Web Development	4	60
BCA7107P	Advance We Development – LAB	2	60
DSE Group–C1			
BCA7108T	Fundamental of Artificial Intelligence	4	60
BCA7108P	Artificial Intelligence – LAB	2	60
BCA7109T	Data Analytics Using Excel	4	60
BCA7109P	Data Analytics Using Excel – LAB	2	60
BCA7110T	Dot Net Programming	4	60
BCA7110P	Dot Net Programming - LAB	2	60
BCA7111T	Fundamental of Machine Learning	4	60
BCA7111P	Machine Learning – LAB	2	60
BCA7112T	Software Engineering	4	60
BCA7112P	Software Engineering – LAB	2	60
DSE Group–D1			
BCA7113P	Project Based on Web Technology	18	540

BCA7114P	Project Based on C / C++ / Java / Python	18	540
BCA7115P	Project Based on Dot Net	18	540
BCA7116P	Project Based on Mobile Application Programming (Eg. Android)	18	540
Skill Enhancement Course (SEC)			
Code	Paper	Credits	Total Hours
SEI7333P	R Programming	2	30
SEI7334P	Introduction to MATLAB	2	30
SEIX335P	Digital Marketing LAB	2	30
SEIX336P	Open Source Programming LAB	2	30
SEIX337P	Big Data–LAB	2	30
SEIX338P	Data Mining & Knowledge Management LAB	2	30

BCA I Semester

Code of the course : BCA5000T
Title of the course : Introduction to Information Technology & PC Packages
Level of the Course : NHEQF Level 4.5
Credit of the Course : 4
Type of the Course : Discipline Centric Core Course (DCC) for Computer Science
Delivery Type of the Course : Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.

Prerequisites:

Introduction to Information Technology & PC Packages courses of the bachelor degree course in computer science or equivalent.

Objectives of the Course:

The objectives of the Programme shall be to provide the sound academic base from which an advanced career in Computer Application can be developed. Conceptual grounding in computer usage as well as its practical business application will be provided.

Learning Outcomes: The student would be able to understand:

1. To give basic information about the computer system.
2. To give knowledge about computer hardware and computer software.
3. To familiarize students with the use of MS Windows, Internet and E-mail.
4. To familiarize students with the use of MS Office - MS Word, MS Excel & MS PowerPoint.

UNIT I

Computer Basics and its generations:

A Simple Model of a Computer, Characteristics and classification of Computers, Generations of Computers. Basic Applications of Computer; Components of Computer System, Central Processing Unit (CPU), VDU, Keyboard and Mouse, Other input/output Devices, Computer Memory organization and hierarchy, Concepts of Hardware and Software; Concept of Computing, Data and Information; Applications of ICT.

(12 Lectures)

UNIT II

Operating Systems:

History and Evolution, Main functions of OS, Multitasking, Multiprocessing, Time Sharing, Real Time OS with Examples, DOS: Introduction, FAT, booting process, DOS system files, DOS commands - internal & external.

Windows Operating System: Introduction, versions, Features, Structure, Utilities, Installation of Hardware & Software, Using Scanner, System Tools, Communication,

Sharing Information between computers and programs,

Linux: Introduction, features, Shell, Kernel, basic commands

(12 Lectures)

UNIT III

Understanding Word Processing: Word Processing Basics; Opening and Closing of documents; Text creation and Manipulation; Formatting of text; Table handling; Spell check, language setting and thesaurus; Printing of word document, Mail Merge, Working with references and Review.

(12 Lectures)

UNIT IV

Using Spreadsheet and Presentations:

Basics of Spreadsheet: Manipulation of cells; Named Range, Conditional Formatting, Formulas and Functions; Graphs and Charts, Pivot tables, sorting, filters, advanced filters, What if analysis, Protecting sheet and workbook, Views.

Making Small Presentation:

Basics of presentation software; Creating Presentation; Preparation and Presentation of Slides; Slide Show; Taking printouts of presentation/handouts.

(12 Lectures)

UNIT-V

Introduction to Internet, WWW and Web Browsers:

Basic of Computer networks; LAN, WAN; Concept of Internet; Applications of Internet; connecting to internet; What is ISP; Knowing the Internet; Basics of internet connectivity related troubleshooting, World Wide Web; Web Browsing software's, Search Engines; Understanding URL; Domain name; IP Address; Using e-governance website, Communications and collaboration: Basics of electronic mail; Getting an email account; Sending and receiving emails; Accessing sent emails; Using Emails; Document collaboration; Instant Messaging; Netiquettes.

(12 Lectures)

Text and Reference Books:

1. P.K. Sinha, Fundamentals of Computers, BPB Publications
2. Fundamentals of Computers By R. Thareja, Oxford University Press.
3. Introduction to Information Technology - ITL Education solutions limited, PEARSON

Suggested E-resources.

1. NPTEL
2. Coursera

BCA I Semester

Code of the course	: BCA5000P
Title of the course	: IT and PC Package Lab
Level of the Course	: NHEQF Level 4.5 Credit of the Course: 2
Type of the Course	: Discipline Centric Core Course (DCC) for Computer Science
Delivery Type of the Course	: Practical 60. The student will perform the experiments and submit the record of observations after getting the results.
Prerequisite	: None

Objectives of the Course :

The objectives of the Programme shall be to provide sound academic base from which an advanced career in Computer Application can be developed. Conceptual grounding in computer usage as well as its practical business application will be provided.

Course Learning Outcomes:

To give basic information about the computer system. To give knowledge about computer hardware and computer software. To familiarize students with the use of MS Windows, Internet and E-mail. To familiarize students with the use of MS Office - MS Word, MS Excel & MS PowerPoint.

LIST OF EXPERIMENTS/EXERCISES

1. All commands specified using DOS, windows & Linux
2. Introduction to MS-Word: Introduction to Word Processing and its features.
3. MS Word: Formatting documents, Paragraph Formatting, Indents.
4. Page formatting, Header and Footer, Bullets and Numbering.
5. Ms-Word: Tabs, Tables, Formatting the tables, Finding and Replacing Text, Mail Merging etc.
6. Introduction to MS-Excel, Introduction to Electronic Spreadsheets, Feature of MS-Excel
7. Entering data, Entering Series, Editing Data, Cell Referencing, Ranges, Formulae, Functions, Auto Sum, Copying Formula
8. MS Excel: Formatting Data, Creating Charts, Creating Database, Sorting Data, Filtering etc.
9. Introduction to MS PowerPoint, PowerPoint, Features of MS-PowerPoint, Slide Animation, Slideshows, Formatting etc.
10. MS-PowerPoint presentation (10-15 slides) on (i) Evolution of Computers (ii) Search Engines (iii) Generation of Computers (iv) Internet

Suggested Books and References:

1. P.K. Sinha, Fundamentals of Computers, BPB Publications
2. Fundamentals of Computers By R. Thareja, Oxford University Press.
3. Introduction to Information Technology - ITL Education solutions limited, PEARSON

Suggested E-resources:

1. <https://www.tutorialspoint.com/disk-operating-system-commands>
2. <https://www.tutorialspoint.com/unix/unix-useful-commands.htm>
3. <https://www.youtube.com/watch?v=fcbB0nkDik8>
4. <https://www.youtube.com/watch?v=S-nHYzK-BVg>

Scheme of EoSE :

(i) Exercise(s): 45 (ii) Viva Voce: 20 (iii) Evaluation of record book: 15

BCA I Semester

Code of the course : BCA5001T
Title of the course : Problem Solving through C Programming
Level of the Course : NHEQF Level 4.5
Credit of the Course : 4
Type of the Course : Discipline Centric Core Course (DCC) for Computer Science
Delivery Type of the Course: Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.

Prerequisite : None

Objectives of the Course: The students will try to learn:

1. Problem-solving through programming
2. Programming language, programming, reading a set of Data, stepwise refinement, concepts of Loops, Functions, Control structure, Arrays, Structure, Pointer and File concept.
3. To build efficient programs in “C” language essential for future programming and software engineering courses.

Learning Outcomes:

1. This lab work provides hands-on experience for C.
2. Demonstrate the behavior of programs involving the basic programming constructs like control structures, string handling.
3. This course provides adequate knowledge to solve problems in their respective domains.
4. Understand and implement concepts on file Handling.
5. Developing application to solve real-world problem using C
6. Become technology-oriented with the knowledge and ability to develop creative solutions, and better understand the effects of future developments of computer systems and technology on people and society as a whole.
7. Acquire some development experience within a specific field of programming C, through project work.

UNIT-I

Algorithm and algorithm development:

Definition and properties of algorithm, flow chart symbols, conversion of flow chart to language, example of simple algorithms, Introduction to program design, errors – syntax error, runtime error, logic error.

Basics of C – Language: Structure of C program, tokens, Data types, constants, operators and its precedence. (12 Lectures)

UNIT-II

Control Structure: Decision Structure: - Simple if, if–else, if–else–if, nested if, switch case

Loop Control Structure: - while, do while and for; Use of break, go to and continue. Jumps in LOOPS

(12 Lectures)

UNIT-III

Functions: Need for functions, Definition of Functions, declaration and prototypes, Return Values and their Types, Function Calls, Category of Functions, No Arguments and no Return Values, Arguments but no Return values, Arguments with Return Values, No Arguments but Returns a Value, Passing Arrays to Functions, Call by Value and Call by Reference, Recursion

(12 Lectures)

UNIT-IV

Arrays and pointers: One Dimensional array, two dimensional arrays, array handling, passing arrays to functions, arrays and string handling, definition of pointers and its uses, pointer arithmetic

Character Arrays and Strings: Declaring and Initializing String Variables, Reading Strings from Terminal, Writing Strings to Screen, Arithmetic Operations on Characters, String-handling Functions

(12 Lectures)

UNIT-V

Storage classes – auto, external, static, register;

Structures – declaring and accessing elements, array of structure,

File Handling - Input/Output, Create, Open, Read, Write, Delete, Close;

(12 Lectures)

Text and Reference Books

1. The C Programming Language, Brian Kernighan and Dennis Ritchie, PHI Publications.
2. Let us C, Yashavant Kanetkar, BPB Publications.
3. Programming in C, Balaguruswamy, McGraw-Hill Education.

Suggested E-resource:

1. <https://www.tutorialspoint.com/cprogramming/index.htm>
2. <https://www.w3schools.com/c/>
3. <https://www.javatpoint.com/c-programming-language-tutorial>
4. <https://www.programiz.com/c-programming>
5. <https://www.nptel.ac.in/courses/108106073/>

BCA I Semester

Code of the course	: BCA5001P
Title of the Course	: Problem Solving through C Programming Lab
Level of the Course	: NHEQF Level 4.5
Credit of the Course	: 2
Type of the Course	: Discipline Centric Core Course (DCC) for Computer Science
Delivery Type of the Course	: Practical 60. The student will perform the experiments and submit the record of observations after getting the results.
Prerequisite	: None

Objectives of the course :

1. To familiarize with C programming constructs.
2. To develop programs in C using basic constructs.
3. To develop programs in C using arrays.
4. To develop applications in C using strings, pointers, functions.
5. To develop applications in C using structures.
6. To develop applications in C using file processing
7. To make the student learn a programming language.
8. To learn problem-solving techniques.
9. To teach the student to write programs in C and to solve the problems.

Learning Outcomes: After Completion of this course the student would be able to

1. Read, understand and trace the execution of programs written in C language.
2. Write the C code for a given algorithm.
3. Implement Programs with pointers and arrays, perform pointer arithmetic, and use the pre-processor.
4. Write programs that perform operations using derived data types.

LIST OF EXPERIMENTS:

1. I/O statements, operators, expressions
2. decision-making constructs: if-else, goto, switch-case, break-continue
3. Loops: for, while, do-while
4. Arrays: 1D and 2D, Multi-dimensional arrays, traversal
5. Strings: operations
6. Functions: call, return, passing parameters by (value, reference), passing arrays to function.
7. Recursion
8. Pointers: Pointers to functions, Arrays, Strings, Pointers to Pointers, Array of Pointers
9. Structures: Nested Structures, Pointers to Structures, Arrays of Structures and Unions.
10. Files: reading and writing, File pointers, file operations, random access, processor directives.

Suggested Programs:

Basics

1. Write a program to print sample strings like “hello world”, “Welcome to C Programming” with different formats using escape sequences.
2. Write a Program to print different datatypes in ‘C’ and their ranges.
3. Write a Program to initialize, assignment & printing variables of different datatypes.

Operators

1. Write a program to demonstrate arithmetic operators. (+, -, *, /, %)
2. Write a Program to demonstrate logical operators. (logical AND, logical OR)
3. Write a Program to read radius value from the keyboard and calculate the area of circle and print the result in both floating and exponential notation.
4. Write a Program to calculate simple interest.
5. Write a Program to convert temperature. (Fahrenheit – Centigrade and vice-versa)

- Write a Program to demonstrate relational operators. (<, >, <=, >=, ==, !=)
- Write a program to check equivalence of two numbers using conditional operator.
- Write a Program to demonstrate pre-increment and post-increment. (++a, a++ where a is a value to be initialized)
- Write a Program to demonstrate pre-decrement and post-decrement. (--a, a-- where a is a value to be initialized)
- Write a program for computing the volume of sphere, cone and cylinder assume that dimensions are integer's use typecasting wherever necessary.

Decision Statements

- Write a Program to read marks of a student in six subjects and print whether pass or fail (using if-else).
- Write a Program to calculate roots of quadratic equation (using if-else).
- Write a Program to calculate electricity bill. Read starting and ending meter reading.
The charges are as follows.
No. of Units Consumed Rate in (Rs)1-100 1.50 per unit
101-300 2.00 per unit for excess of 100 units
301-500 2.50 per unit for excess of 300 units501-above3.25 per unit for excess of 500 units

Switch operations

- Write a Program to perform arithmetic operations using switch case.
- Write a Program to display colors using switch case (VIBGYOR).
- Write a Program to display vowels and consonants using switch case.
- Write a Program to display names of days in a Week using switch case.

Basic Loop Operations-Do the Following Programs Using for, while, do-while loops.

- Write a program to calculate sum of individual digits of a given number.
- Write a program to check whether given number is a palindrome or not.
- Write a program to print prime numbers in the given range.
- Write a program to display multiplication tables from 1 to 10 except 3 and 5.

Advanced loops

- Write a program to print the Fibonacci series for given 'N' value.
- Write a program to check whether a given number is a Fibonacci number or not.
- Write a program to read 2 numbers x and n then compute the sum of the Geometric Progression.
 $1+x+x^2+x^3+ \dots +x^n$
- Write a program to print the following format.

```

1
12
12 3
12 34

```

Arrays

- Write a program to store 10 elements in the 1-D array and print sum of the array.
- Write a program to print minimum and maximum elements in the 1-D array.
- Write a program to count no. of positive numbers, negative numbers and zeros in the array.
- Write a program to search the given element by using linear search.
- Write a program to sort the given elements using bubble sort technique.

Arrays

- Write a program to perform matrix addition and matrix subtraction.
- Write a program to perform matrix multiplication by checking the compatibility.
- Write a program to print the transpose of a matrix.

Strings

- Write a program to perform various string manipulations using built-in functions.
- Write a program to print the given strings in ascending order.
- Write a program to verify the given string is palindrome or not (without built-in functions, with using built-in functions).
- Write a program to concatenate two strings using arrays.

Math Functions and I/O Functions

1. Write a program to read values from keyboard and find the values using `abs()`, `sqrt()`, `floor()`, `ceil()` and `pow()`.
2. Write a program to read and display a value using `getch()` and `putch()`.
3. Write a program to read and display a value using `getchar()`, `putchar()`, `gets()` and `puts()`.

Functions

1. Write a program to find sum of two numbers using functions.
2. Write a program to find product of two numbers using functions without arguments, without return type.
3. Write a program to find difference of two numbers using functions without arguments, with return type.
4. Write a program to find sum of two numbers using functions with arguments & without return type.
5. Write a program to find product of two numbers using functions with arguments, with return type.

Functions and Recursion

1. Write a program to swap two numbers using
2. Call By Value B) Call By Reference.
3. Write a program to calculate factorial, gcd using recursion and non-recursion functions.
4. Write program to perform arithmetic operations using pointer.
5. Write a program for matrix addition using pointers.

Structures

1. Write a program to create structure for an account holder in a bank with following Fields: name, account number, address, balance and display the details of five account holders.
2. Write a program to find total marks of individual student and average marks for 10 students using structures.
3. Write a program to create structure called traveler and members of structure are train no, coach no, seat no, source, destination, gender, age, name and departure date.
4. Write a program to illustrate passing an entire structure to a function.

File Operations using Command Line Arguments

1. Write a program which copies the contents of one file to another file using command line arguments.
2. Write a program to reverse the first n characters in a file using command line arguments.

Text and Reference Books:

1. Problem Solving and Program Design in C, 4th edition, by Jeri R. Hanly and Elli B. Koffman.
2. Programming in C by Pradip Dey, Manas Ghosh 2nd edition, Oxford University Press.
3. E. Balaguruswamy, Programming in ANSI C, 5th Edition, McGraw-Hill.
4. A first book of ANSI C by Gray J. Brosin 3rd edition, Cengage Delmar Learning India Pvt. Ltd.
5. AL Kelly, Iraphol, Programming in C, 4th edition, Addison-Wesley–Professional.
6. Brian W. Kernighan & Dennis Ritchie, C Programming Language, 2nd edition, PHI.

Suggested E-resource:

1. Tutorialspoint C Programming](<https://www.tutorialspoint.com/cprogramming/index.htm>)
2. W3Schools C](<https://www.w3schools.com/c/>)
3. JavaTpoint C Programming Language Tutorial](<https://www.javatpoint.com/c-programming>)
4. Programiz C Programming](<https://www.programiz.com/c-programming>)
5. NPTEL C Courses](<https://www.nptel.ac.in/courses/108106073/>)

Scheme of EoSE :

(i) Exercise(s):45 (ii) Viva Voce: 20 (iii) Evaluation of record book: 15

BCA I Semester

Code of the course	: BCA5002T
Title of the course	: Database Management System
Level of the Course	: NHEQF Level 4.5
Credit of the Course	: 4
Type of the Course	: Discipline Centric Core Course (DCC) for Computer Science
Delivery Type of the Course	: Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.

Prerequisite:

1. Although DBMS is easy to understand, it is recommended to have fundamental computer knowledge that includes concepts of computer architecture, storage and hardware.
2. Knowledge of data structures and algorithms and programming will be an added benefit.

Objectives of the Course:

1. Understand the role of database management system in an organization and learn the database Concepts.
2. Design databases using data modeling and Logical database design techniques.
3. Construct database queries using relational algebra and calculus and SQL.
4. Understand the concept of a database transaction and related concurrent, recovery facilities.
5. Learn how to evaluate a set of queries in query processing

Learning Outcomes:

1. Define the terminology, features, and characteristics of database system.
2. Differentiate database systems from file systems by enumerating various features provided by database systems.
3. Describe Data Models, Schemas, Instances, Three Schema Architecture and DBMS Component Modules.
4. Analyze an information storage problem and derive an information model expressed in the form of an entity relation diagram.
5. Model the real-world database systems using Entity Relationship Diagrams (ERD) from the requirements specification.
6. Describe basics of the relational data model.
7. Define and illustrate the Relational Data Model, Constraints and Schemas.
8. Transform an information model into a relational database schema and implement schema using data definition language and/or utilities.
9. Formulate solutions to a broad range of query problems using relational algebra.
10. Illustrate the Functional Dependencies, Inference Rules, Minimal Sets of FDs.
11. Understand normalization theory and criticize a database design and improve the design by normalization.
12. Explain various Normal Forms and Apply to normalize a database.
13. Understand the SQL Data definition statements to formulate solutions to a broad range of query and data update problems.
14. Use SQL queries for data aggregation, calculations, views, sub-queries, embedded queries, manipulation
15. Define disk storage devices, files of records, unordered files, ordered files and hashed files and organizations.
16. Familiar with basic database storage structures and access techniques- file and page organizations, indexing methods.
17. Illustrate various operations in implementing data indices using various hashing techniques.

UNIT-I

Introduction: Purpose of the data base system, data abstraction, data model, data independence, data definition language, data manipulation language, data base administrator, data base users, overall structure.

ER Model: entities, mapping constrains, keys, E-R diagram, reduction E-R diagrams to tables, specialization, generalization, aggregation, design of an E-R database scheme.

(12 Lectures)

UNIT-II

Relational Model: The catalog, base tables and views. Relational Data Objects - Domains and Relations: Domains, relations, kinds of relations, relations and predicates, relational databases.

Relational Data Integrity - Candidate keys and related matters: Candidate keys.

Primary and alternate keys. Foreign keys, foreign key rules, nulls. Candidate keys and nulls, foreign key and nulls.

(12 Lectures)

UNIT-III

Structured Query Language: Create relationships between database tables, auto increment, check, Null values, aggregate functions - min, max, count, average, sum, nested sub-queries, group by, having, exists, case, order by. Join operations - inner, left join, right join, natural join and Cartesian product.

Views: Introduction, what are views for, data definition, data manipulation, SQL support.

(12 Lectures)

UNIT-IV

Network model: Basic concepts, data structure diagrams, DBTG CODASYL model, DBTG data retrieval facility, DBTG update facility, DBTG set processing facility, mapping networks to file, networks system.

Hierarchical model: basic concepts, tree structure diagrams, data retrieval facility, update facility, virtual records, mapping hierarchical to files, hierarchical system.

(12 Lectures)

UNIT-V

Normalization: Functional dependencies, normal forms- First, second, third, BCNF, fourth and fifth normal forms

File and system structure: overall system structure, file organization, logical and physical file organization, sequential and random, hierarchical, inverted, null list, indexing and hashing, B-tree index files.

(12 Lectures)

Text and Recommended Books

1. Fundamentals of Database System, Shamkant B. Navathe, Ramez Elmasri, Pearson.
2. Korth, Database Systems Concepts, McGrawHill.
3. Date C.J., Database Systems, AddisonWesley

Suggested E-resources:

1. https://www.youtube.com/results?search_query=DBMS+online+classes
2. <http://www.w3schools.in/dbms/>
3. <http://beginnersbook.com/2015/04/dbms-tutorial>

BCA I Semester

Code of the course	: BCA5002P
Title of the course	: Database Management System Lab
Level of the Course	: NHEQF Level 4.5
Credit of the Course	: 2
Type of the Course	: Discipline Centric Core Course (DCC) for Computer Science
Delivery Type of the Course	: Practical 60. The student will perform the experiments and submit the record of observations after getting the results.

Prerequisites :

1. You should have a basic understanding of Computer Programming terminologies. Must have knowledge.
2. A basic understanding of any of the programming languages will help you in understanding the Math concepts and move fast on the learning track.

Objectives of the Course:

1. Keep abreast of current developments to continue their own professional development.
2. To engage themselves in lifelong learning of Database management systems practical's and technologies this enables them to pursue higher studies.
3. To interact professionally with colleagues or clients located abroad and the ability to overcome challenges that arises from geographic distance, cultural differences, and multiple languages in the context of computing.
4. Develop team spirit, effective work habits, and professional attitude in written and oral forms, towards the development of database applications

Learning Outcomes:

1. Define the terminology, features, and characteristics of database system.
2. Differentiate database systems from file systems by enumerating various features provided by database systems.
3. Describe Data Models, Schemas, Instances, Three Schema Architecture and DBMS Component Modules.
4. Analyze an information storage problem and derive an information model expressed in the form of an entity relation diagram.
5. Model the real-world database systems using Entity Relationship Diagrams (ERD) from the requirements specification.
6. Transform an information model into a relational database schema and implement schema using data definition language and/or utilities.
7. Formulate solutions to a broad range of query problems using relational algebra.
8. Illustrate the Functional Dependencies, Inference Rules, Minimal Sets of FDs.
9. Understand normalization theory and criticize a database design and improve the design by normalization.
10. Explain various Normal Forms and Apply to normalize a database.
11. Understand the SQL Data definition statements to formulate solutions to a broad range of query and data update problems.
12. Use SQL queries for data aggregation, calculations, views, sub-queries, embedded queries, manipulation
13. Familiar with basic database storage structures and access techniques- file and page organizations, indexing methods.
14. Illustrate various operations in implementing data indices using various hashing techniques.

LIST OF EXPERIMENTS/EXERCISES

1. How to create database.
2. How to create table in database.
3. How to insert data in table.
4. Create new table from existing table.
5. How to drop database.
6. How to delete a table.
7. How to display all data from table.
8. How to display specific data from table.
 - a. Alias Rename table name
 - b. Rename column name with data type.
9. How to add column in existing table.
10. Rename column name using Alter table.
11. Change column size using alter table.
12. Change column data type using alter table.
13. Update data using update statement.
14. Update old data with new data.
15. Fetch data using condition.
16. Delete data using condition.
17. Drop column in table.
18. Increase salary of all employees using single update.
19. Concat multiple column in one column.
20. +, -, *, / operators.
21. Fetch data using built in functions.
22. Group by clause.
23. Order by clause.
24. Identity Constraints (PK, FK, check, unique, default, not null).
25. Joins.
26. Views.
27. How to create index.
28. Pattern matching-Like operator.
29. Between operator.
30. In, Not In operator.
31. In operator with sub query.
32. Top operator.
33. Fetch and delete Null and Not Null data from table.
34. Distinct operator.
35. Truncate, Rollback, Commit

Suggested Books and References:

1. “Database Systems, Concepts, Design and Applications” by S.K. Singh, Pearson Education.
2. “Database Management Systems” by Raghu Ramakrishnan, Johannes Gehrke, McGraw Hill Publication.
3. “Fundamentals of Database Systems” by Elmsari, Navathe, 5th Edition, Pearson Education (2008).

Suggested E-resources:

1. NPTEL
2. Coursera

Scheme of EoSE :

- (i)Exercise(s):45 (ii)Viva Voce: 20 (iii) Evaluation of record book: 15

BCA II SEMESTER

Code of the course	: BCA5003T
Title of the course	: Object Oriented Programming C++
Level of the Course	:NHEQF Level 4.5
Credit of the Course	: 4
Type of the Course	: Discipline Centric Core Course (DCC) for Computer Science
Delivery Type of the Course	: Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.

Prerequisites : Study C Programming concepts for this programming subject.

Objectives of the Course:

Object-oriented programming (OOP) is a way of thinking about and organizing code for maximum reusability. With this type of programming, a program comprises objects that can interact with the user, other objects, or other programs. This makes programs more efficient and easier to understand.

Learning Outcomes:

1. Understand the principles and concepts of Object-Oriented Programming (OOP) and its advantages over procedural programming. Differentiate between OOP and procedural programming paradigms. Apply the principles of abstraction, encapsulation, inheritance, and polymorphism in problem-solving.
2. Demonstrate proficiency in C++ programming language basics, including the structure of a C++ program, data types, variables, expressions, operators, pointers, arrays, strings, flow control statements, functions, and memory management. Implement recursive functions, inline functions, and dynamic memory allocation.
3. Design and implement classes and data abstraction in C++. Understand class definition, objects, scope, constructors, destructors, static members, constant member functions, friend functions, and dynamic creation and destruction of objects. Apply function overloading and operator overloading techniques.
4. Explore inheritance and polymorphism concepts in C++. Define class hierarchies, implement different forms of inheritance, and understand base and derived class relationships. Utilize access specifiers to control access to base class members. Implement virtual functions and understand the dynamic binding mechanism. Use abstract classes and handle polymorphic use of classes effectively.
5. Gain proficiency in advanced C++ topics, including templates and exception handling. Implement function templates and class templates, and understand overloading in templates. Handle exceptions in C++ programs, including throwing exceptions, catching exceptions, re-throwing exceptions, and specifying exception specifications. Utilize file handling techniques for stream I/O, file operations, error handling, and formatted I/O. Understand the components of the Standard Template Library (STL) and their applications.

UNIT – I

Different paradigms for problem solving, need for OOP, differences between OOP and procedure-oriented programming, abstraction, overview of OOP principles- encapsulation, inheritance and data binding polymorphism, abstraction.

C++ basics: structure of a C++ program, data types, declaration of variables, expressions, operators, type conversions, pointers and arrays, strings, structures, references, flow control statement, functions-scope of variables, parameter passing, recursive functions, default arguments, inline functions, dynamic memory allocation and deallocation operators.

(12 Lectures)

UNIT – II

C++ classes and data abstraction: class definition, class structure, class objects, class scope, this pointer, static class members, constant member functions, constructors and destructors, dynamic creation and destruction of objects, friend function and class, static class member

Overloading: function overloading, operator overloading – unary, binary operators.

(12 Lectures)

UNIT – III

Inheritance: defining a class hierarchy, different forms of inheritance, defining the base and derived classes, access to the base class members, base and derived class construction, destructors, virtual base class.

Polymorphism: static and dynamic bindings, base and derived class virtual functions, dynamic binding through virtual functions, virtual function call mechanism, pure virtual functions, abstract classes, implications of polymorphic use of classes, virtual destructors.

(12 Lectures)

UNIT – IV

Templates - function templates and class templates, overloading of function template, static class member in class template.

Exception handling: benefits of exception handling, throwing an exception, the try block, catching an exception, exception objects, exception specifications, rethrowing an exception, catching all exceptions.

(12 Lectures)

UNIT-V

File handling: stream classes hierarchy, stream I/O, file streams, opening and closing datafile, creating a data file, read and write functions, error handling during file operations, formatted I/O, sequential and random file processing.

Standard template library (STL): component of STL, containers, iterators, algorithms, application of container classes.

(12 Lectures)

Test and References Books:

1. Object Oriented Programming with C++: E. Balagurusamy
2. C++: The Complete Reference, Herbert Schildt, McGraw Hill.
3. Let Us C++, Yashwant Kanetkar, Bpb Publisher.
4. C, C++ C Programming Concepts Black Book, Dasgupta, Chakrabarti, Dreamtech Press

Suggested E-resources.

1. NPTEL
2. Coursera

BCA II SEMESTER

Code of the course	: BCA5003P
Title of the course	: Object Oriented Programming C++ Lab
Level of the Course	: NHEQF Level 4.5
Credit of the Course	: 2
Type of the Course	: Discipline Centric Core Course (DCC) for Computer Science
Delivery Type of the Course	: Practical 60. The student will perform the experiments and submit the record of observations after getting the results.
Prerequisites	: Study C Programming concepts for this programming subject.

Objectives of the Course:

Object-oriented programming (OOP) is a way of thinking about and organizing code for maximum reusability. With this type of programming, a program comprises objects that can interact with the user, other objects, or other programs. This makes programs more efficient and easier to understand.

Learning Outcomes:

1. Understand the principles and concepts of Object-Oriented Programming (OOP) and its advantages over procedural programming. Differentiate between OOP and procedural programming paradigms. Apply the principles of abstraction, encapsulation, inheritance, and polymorphism in problem-solving.
2. Demonstrate proficiency in C++ programming language basics, including the structure of a C++ program, data types, variables, expressions, operators, pointers, arrays, strings, flow control statements, functions, and memory management. Implement recursive functions, inline functions, and dynamic memory allocation.
3. Design and implement classes and data abstraction in C++. Understand class definition, objects, scope, constructors, destructors, static members, constant member functions, friend functions, and dynamic creation and destruction of objects. Apply function overloading and operator overloading techniques.
 - a. Explore inheritance and polymorphism concepts in C++. Define class hierarchies, implement different forms of inheritance, and understand base and derived class relationships. Utilize access specifiers to control access to base class members. Implement virtual functions and understand the dynamic binding mechanism. Use abstract classes and handle polymorphic use of classes effectively.
4. Gain proficiency in advanced C++ topics, including templates and exception handling. Implement function templates and class templates, and understand overloading in templates. Handle exceptions in C++ programs, including throwing exceptions, catching exceptions, re-throwing exceptions, and specifying exception specifications. Utilize file handling techniques for stream I/O, file operations, error handling, and formatted I/O. Understand the components of the Standard Template Library (STL) and their applications.

LIST OF EXPERIMENTS/EXERCISES: -

1. Write a program for Basic/Simple logic building in C++
2. Write a program to implement the concept of class and object
3. Write a program to implement the concept of Inline function.
4. Write a program to implement function overloading
5. Write a program to implement the concept of friend function
6. Write a program to implement the concept of static data member.
7. Write a program to implement static member function.
8. Write a program using the concept of constructor.
9. Write a program using the concept of constructor.

10. Write a program to implement operator overloading
11. Write a program to implement single inheritance
12. Write a program to Implement Multiple inheritance
13. Write a program to Implement Virtual function
14. Write a program to implement Virtual Base class
15. Write a program to create, read & write sequential file
16. Write a program to create, read & write random access file
17. Write a program to enter 5 records into a file and display them.
18. Write a program to implement function template
19. Write a program to implement class template
20. Write a program for exception handling

Suggested Books and References:

1. Object Oriented Programming with C++: E. Balagurusamy
2. C++: The Complete Reference, Herbert Schildt, McGraw Hill.
3. Let Us C++, Yashwant Kanetkar, Bpb Publisher.
4. C, C++ C Programming Concepts Black Book, Dasgupta, Chakrabarti, Dreamtech Press

Suggested E-resources.

1. NPTEL
2. Coursera

Scheme of EoSE :

(i)Exercise(s):45

(ii)Viva Voce: 20

(iii) Evaluation of record book: 15

BCA II SEMESTER

Code of the course : BCA5004T
Title of the course : Data Structure
Level of the Course : NHEQF Level 4.5
Credit of the Course : 4
Type of the Course : Discipline Centric Core Course (DCC) for Computer Science
Delivery Type of the Course : Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.

Prerequisites : Problem Solving using C Code or equivalent.

Objectives of the Course :
Storing data while maintaining the data's correctness and efficiency in a computer program.

Course Learning Outcomes:

1. Understand and apply linear data structures, such as arrays, records, stacks, and queues, including their operations and implementations using arrays.
2. Demonstrate proficiency in linked data structures, specifically linked lists, including operations like insertion, deletion, and traversal. Understand and implement circular linked lists and doubly linked lists.
3. Gain knowledge of tree structures, particularly binary search trees, and effectively perform operations like insertion, deletion, and searching. Implement algorithms for these operations and understand tree traversals.
4. Develop an understanding of graph structures and their representations using adjacency matrix, adjacency list, and adjacency multilist. Perform graph traversals using BFS and DFS. Explore concepts such as shortest path, transitive closure, and reflexive transitive closure.
5. Acquire skills in searching and sorting algorithms, including sequential search, binary search, hashing, selection sort, bubble sort, quick sort, heap sort, merge sort, and insertion sort. Consider efficiency considerations while implementing and analyzing these algorithms.
6. Students will be able to demonstrate their skills

UNIT-I

Linear Structure: Arrays, records, stack, operations on stack, implementation of stack as an array, queue, operations on queue, implementation of queue.

(12 Lectures)

UNIT-II

Linked Structure: List representation, operations on linked list - get node and free node operation, implementing the list operation, inserting into an ordered linked list, deleting, circular linked list, doubly linked list.

(12 Lectures)

UNIT-III

Tree Structure: Binary search tree, inserting, deleting, and searching into binary search tree, implementing the insert, search and delete algorithms, tree traversals.

(12 Lectures)

UNIT-IV

Graph Structure: Graph representation - Adjacency matrix, adjacency list, adjacency multilist representation. Orthogonal representation of graph. Graph traversals - BFS and DFS. Shortest path, all pairs of shortest paths, transitive closure, reflexive transitive closure.

(12 Lectures)

UNIT-V

Searching and sorting: Searching - sequential searching, binary searching, hashing. Sorting - selection sort, bubble sort, quick sort, heap sort, merge sort, and insertion sort, efficiency considerations.

(12 Lectures)

Suggested Books and References:

1. Horowitz, Sahni, Fundamentals of Data Structure, Galgotia Publication Private Limited
2. Data Structure Using C & C++, Tannenbaum, PHI
3. Data Structure, Lipschutz, Tata McGraw-Hill Education Private Limited

Suggested E-resources:

1. NPTEL
2. Coursera

BCA II SEMESTER

Code of the course	: BCA5004P
Title of the course	: Data Structure Lab
Level of the Course	: NHEQF Level 4.5
Credit of the Course	: 2
Type of the Course	: Discipline Centric Core Course (DCC) for Computer Science
Delivery Type of the Course	: Practical 60. The student will perform the experiments and submit the record of observations after getting the results.
Prerequisites	: Problem Solving using C courses of bachelor degree course

Objectives of the Course :

The course allows us in storing data while maintaining the data's correctness and efficiency in a computer program.

Learning Outcomes:

1. Understand and apply linear data structures, such as arrays, records, stacks, and queues, including their operations and implementations using arrays.
2. Demonstrate proficiency in linked data structures, specifically linked lists, including operations like insertion, deletion, and traversal. Understand and implement circular linked lists and doubly linked lists.
3. Gain knowledge of tree structures, particularly binary search trees, and effectively perform operations like insertion, deletion, and searching. Implement algorithms for these operations and understand tree traversals.
4. Acquire skills in searching and sorting algorithms, including sequential search, binary search, selection sort, bubble sort, quick sort, heap sort, merge sort, and insertion sort.

List of Programs:

1. Write a C program to find minimum number in 1-D array.
2. Write a C program to find maximum number in 1-D array.
3. Write a C program to perform arithmetic operations on 1-D array.
4. Write a C program to perform addition and subtraction operations on 2-D array.
5. Write a C program to perform multiplication operation on 2-D array.
6. Write a C program to perform transpose of matrix.
7. Write a C program to implement stack using array.
8. Write a C program to implement queue using array.
9. Write a C program to implement circular queue using array.
10. Write a C program to perform insertion and traverse operations in singly linked list.
Perform following operations:
 - a. Create a list
 - b. Traverse/display
 - c. Insert at begin
 - d. Insert at end
 - e. Insert after given INFO node
11. WAP to perform deletion and traverse operations in singly linked list. Perform following operations:
 - a. Create a list
 - b. Traverse/display
 - c. Delete from begin
 - d. Delete from end
 - e. Delete if information of node is given
12. WAP to perform insertion and traverse operations in doubly linked list. Perform following operations:
 - a. Create a doubly linked list
 - b. Forward traverse

- c. Backward traverse
 - d. Insert at begin
 - e. Insert at end
 - f. Insert after given INFO node
13. WAP to perform deletion and traverse operations in doubly linked list. Perform following operations:
- a. Create a doubly linked list
 - b. Forward traverse
 - c. Backward traverse
 - d. Delete from begin
 - e. Delete from end
 - f. Delete given INFO node
14. Write a program to implement Linear Search.
15. Write a program to implement Binary Search.
16. Write a program to implement Bubble Sort.
17. Write a program to implement Insertion Sort.
18. Write a program to implement Selection Sort.
19. Write a program to implement Quick Sort.
20. Write a program to implement Merge Sort.
21. Write a program to implement Heap Sort.
22. Write a program to implement Binary Search Tree.
23. Write a program to implement sparse matrix.
24. Write a program in C to evaluate a valid postfix expression using stack. Assume that the postfix expression is read as a single line consisting of non-negative single-digit operands and binary arithmetic operators. The operators are + (add), - (subtract), (multiply), / (divide).

Suggested Books and References:

1. Horowitz, Sahni, Fundamentals of Data Structure, Galgotia Publication Private Limited
2. Data Structure Using C & C++, Tannenbaum, PHI
3. Data Structure, Lipschutz, Tata McGraw-Hill Education Private Limited

Suggested E-resources:

1. NPTEL
2. Coursera

Scheme of EoSE: -

- (1) Exercise(s): 45
- (2) Viva Voce: 20
- (3) Evaluation of record book: 15

BCA II SEMESTER

Code of the course	: BCA5005T
Title of the course	: Computer Organization and Architecture (COA)
Level of the Course	: NHEQF Level 4.5
Credit of the Course	: 4
Type of the Course	: Discipline Centric Core Course (DCC) for Computer Science
Delivery Type of the Course	: Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.
Prerequisites	: None

Objectives of the Course: The educational Objectives of this Course are

1. To have a thorough understanding of the basic structure and operation of a digital computer.
2. To study the different ways of communicating with I/O devices and standard I/O interfaces.
3. To learn the architecture and assembly language programming of 8085 microprocessor.
4. To study peripherals and their interfacing with 8085 microprocessors.

Learning Outcomes:

1. Ability to understand the basic structure of the computer.
2. Ability to perform computer arithmetic operations.

UNIT I

Introduction to Computer Organization: Basic building blocks of computer system, Stored Program Control Concept.

Data Representation: Understanding bits and bytes, and the binary number system. Conversion between binary, decimal, octal, and hexadecimal. Representing integers, floating-point numbers.

Floating Point Arithmetic: Addition, Subtraction, Multiplication, and Division Algorithms.

(12 Lectures)

UNIT II

Overview and Instructions: Instruction Code, Computer registers, Instruction Cycle, Memory reference instruction, Flynn's classification of computers.

Addressing modes: register, immediate, direct, indirect, and indexed.

Operations in the instruction set: Arithmetic, Logical, Data Transfer, and Control Flow.

(12 Lectures)

UNIT III

Memory Organisation: Classification of Memory and Hierarchy.

Cache basics: Measuring and improving cache performance, Introduction to Virtual memory, Translation Lookaside Buffers (TLBs), Input/output system, programmed I/O, DMA and Interrupts, I/O processors.

(12 Lectures)

UNIT IV

Introduction to Parallel Processing: Pipelining, Arithmetic and Instruction Pipeline, Characteristics of multiprocessors, Interconnection structures, Inter-processor Arbitration, Inter-processor Communication and Synchronization.

Pipelined datapath and Pipelined control, Handling Data hazards and Control hazards.

(12 Lectures)

UNIT V

Instruction Set Architecture: Reduced Instruction Set Computer (RISC), Complex Instruction Set Computer (CISC), and their comparison.

Design of Control Unit: Hardwired and Micro-programmed Control Unit.

Introduction to Microprocessors 8085: Features, Bus Structure (Data, Address, Control bus), Instruction Set of 8085, Introduction to 8085 Assembly Language Programming.

(12 Lectures)

Text and Book References:

1. System Architecture, M. Morris R., Mano, PHI
2. Computer Computer Organization and Design, David A. Patterson and John L. Hennessy, Morgan

Kauffman.

3. Microprocessor, Architecture, Programming, & Applications with the 8085 6/e (Ramesh Gaonkar)
PENRAM INTL.
4. Computer Architecture and Organization, John P. Hayes, TMH.
5. Structured Computer Organization, Andrew S. Tanenbaum and Todd Austin, PHI.

Suggested E-resources:

1. Introduction to Computer Organization and Architecture COA (nitsri.ac.in)
2. CAO.pdf (citchennai.edu.in)
3. Parallel Processing - javatpoint

BCA II SEMESTER

Code of the course	: BCA5005P
Title of the course	: Computer Organization and Architecture Lab
Level of the Course	: NHEQF Level 4.5
Credit of the Course	: 2
Type of the Course	: Discipline Centric Core Course (DCC) for Computer Science
Delivery Type of the Course	: Practical 60. The student will perform the experiments and submit the record of observations after getting the results.
Prerequisites	: None

Objectives of the Course: The educational Objectives of this Course are

1. To have a thorough understanding of the basic structure and operation of a digital computer.
2. To study the different ways of communicating with I/O devices and standard I/O interfaces.
3. To learn the architecture and assembly language programming of 8085 microprocessor.
4. To study peripherals and their interfacing with 8085 microprocessors.

Learning Outcomes: At the end of the course the student would:

1. Ability to understand the basic structure of the computer.
2. Ability to perform computer arithmetic operations.

List of Experiments

1. INTRODUCTION TO 8085 MICROPROCESSORS
2. 1's complement of 8-bit numbers
3. 2's complement of 8-bit numbers
4. 1's complement of 16-bit numbers
5. 2's complement of 16-bit numbers
6. Shift left 8-bit number by 1 bit
7. Shift right 8-bit number by 1 bit
8. Mask the lower nibble of an 8-bit number
9. Mask the higher nibble of an 8-bit number
10. Add two 8-bit numbers without considering the carry
11. Add two 8-bit numbers with considering the carry
12. Subtract two 8-bit numbers without considering the borrow
13. Subtract two 8-bit numbers with considering the borrow
14. Add two 16-bit numbers without considering the carry
15. Subtract two 16-bit numbers without considering the borrow
16. Add two 8-bit numbers and show the result in the decimal number system
17. Multiply two 8-bit numbers
18. Find the square of an 8-bit number
19. Larger of two 8-bit numbers
20. Smaller of two 8-bit numbers.
21. Addition of ten 8-bit numbers stored in memory
22. Find the number of negative elements in a block of data
23. To sort numbers
24. Alter the contents of the flag register in 8085
25. Calculate the sum of a series of numbers
26. Division of 16-bit numbers by 8-bit numbers
27. Find the number of negative elements
28. Find the largest of given numbers
29. Count the number of 1's in the content of a register
30. Transfer content to overlapping memory blocks
31. Find the factorial of 8-bit numbers.
32. Find the Positive, negative, or zero numbers in elements
33. Conversion -> BCD to HEX, HEX to BCD

34. Calculate the sum of a series of even numbers
35. Calculate the sum of a series of odd numbers
36. Exchange a block of N numbers
37. Pack the unpack BCD numbers
38. Unpack the pack BCD numbers
39. Arrange an array of data in ascending orders
40. Arrange an array of data in descending orders
41. Show the smallest among two numbers
42. Show the smallest and largest two numbers

Scheme of EoSE: -

- (1) Exercise(s): 45
- (2) Viva Voce: 20
- (3) Evaluation of the record book: 15

BCA III SEMESTER

Code of the course : BCA6006T
Title of the course : JAVA Programming
Level of the Course : NHEQF Level 5.0
Credit of the Course : 4
Type of the Course : Discipline Centric Core Course (DCC) in computer science
Delivery Type of the Course : Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem-solving.

Prerequisite:

1. C language skills (Good to have)
2. This course is designed to meet the needs of those who want to be professional Java developers

Objectives of the Course:

1. To become familiar with the features of the Java Language
2. To discover how to write Java code according to Object-Oriented Programming principles.
3. To become comfortable with concepts such as Classes, Objects, Inheritance, Polymorphism, and Interfaces
4. To learn Java APIs for Collections, I/O Streams
5. To design GUI applications and Applets using AWT and Swing.
6. To develop Multithreaded and Networking applications.

Learning Outcomes:

1. Use object-oriented programming concepts to solve real-world problems.
2. Explain the concept of class and objects with access control to represent real-world entities.
3. Demonstrate the behavior of programs involving the basic programming constructs like control structures, constructors, string handling, and garbage collection.
4. Use overloading methodology on methods and constructors to develop application programs.
5. Demonstrate the implementation of inheritance (multilevel, hierarchical, and multiple) by using extend and implement keywords.
6. Describe the concept of interface and abstract classes to define generic classes.
7. Use dynamic and static polymorphism to process objects depending on their class.
8. Illustrate different techniques for creating and accessing packages (fully qualified name and import statements).

UNIT-I

Introduction to Java: Bytecode, features of Java, datatypes, variables and arrays, operators, control statements.

Objects & Classes: Object Oriented Programming, defining classes, static fields and methods, object construction

(12 Lectures)

UNIT-II

Inheritance: Basics, using super, method overriding, using abstract classes, using final with inheritance.

Packages and Interfaces: Defining a package, importing package, defining an interface, implementing and applying interfaces.

(12 Lectures)

UNIT-III

Exception Handling: Fundamentals, exception types, using try and catch.

Multithreaded Programming: Creating a single and multiple threads, thread priorities, synchronization.

(12 Lectures)

UNIT-IV

Applets: Applets basics, applets architecture, applets skeleton, the html applet tag, passing parameters in applets.

Event-Driven Programming: Event-Handling Process, Event Handling Mechanism, The Delegation Model of Event Handling, Event Classes, Event Sources, Event Listeners, Adapter Classes as Helper Classes in Event Handling.

(12 Lectures)

UNIT-V

Designing Graphical User Interfaces: Components and Containers, Basics of Components, Using Containers, Layout Managers, AWT Components, Adding a Menu to Window, Extending GUI Features Using Swing Components

(12 Lectures)

Text and Reference Books:

1. The complete reference Java 2, H. Schildt, Tata McGraw-Hill.
2. Programming with JAVA - A Primer, Balaguruswamy, McGraw-Hill
3. Head First Java: A Brain-Friendly Guide, Kathy Bert"O'Reilly Media.
4. Thinking in Java, Bruce, Prentice Hall Professional
5. Learning Java: A Bestselling Hands-On Java Tutorial, Patrick Niemeyer, Daniel Leuck, "O'Reilly Media,
6. How to program in Java, Deitel and Dietel

Suggested E-resources:

1. <http://java.sun.com>
2. <http://www.oracle.com/technetwork/java/index.html>
3. <http://java.sun.com/javase>
4. <http://www.oracle.com/technetwork/java/javase/overview/index.html>
5. <http://download.oracle.com/javase/7/docs/api/index.html>

BCA III SEMESTER

Code of the course	: BCA6006P
Title of the Course	: Java programming Lab
Level of the Course	: NHEQF Level 5.0
Credit of the Course	: 2
Type of the Course	: Discipline Centric Core Course (DCC) in computer science
Delivery Type of the Course	: Practical 60. The student will perform the experiments and submit the record of observations after getting the results.
Prerequisite	: Basics knowledge of Object-oriented concepts of java programming lab

Objectives of the Course:

1. To understand the basic concepts and fundamentals of platform-independent object-oriented language.
2. To demonstrate skills in writing programs using exception handling techniques and multithreading.
3. To understand streams and efficient user interface design techniques.
4. Develop applications using Console I/O and File I/O, GUI application

Learning Outcomes: After successful completion of the course, the students are able to

1. Use the syntax and semantics of java programming language and basic concepts of OOP. Develop reusable programs using the concepts of inheritance, polymorphism, interfaces and packages.
2. Apply the concepts of Multithreading and Exception handling to develop efficient and error-free codes.
3. Design event-driven GUI and web-related applications which mimic the real-world scenarios

LIST OF EXPERIMENTS:

Note: The lab instructor is expected to design problems based on the topics listed. The Examination shall not be restricted to the sample experiments designed.

1. Introduction of OOPs
2. Class and its Objects
3. Inheritance
4. Packages and Interfaces
5. Variables and Inner Classes.
6. Exception handling
7. Java I/O
8. String
9. Multithreading
10. Applet
11. Event Handling
12. Swing and its Component Layout

Suggested Programs:

Practical No. 1 - Java Basics

1. Write a Java program that takes a number as input and prints its multiplication table up to 10.
2. Write a Java program to display the following pattern:

```
*****
****
***
**
*
```

3. Write a Java program to print the area and perimeter of a circle.

Practical No. 2 - Use of Operators

1. Write a Java program to add two binary numbers.
2. Write a Java program to convert a decimal number to a binary number and vice versa.
3. Write a Java program to reverse a string.

Practical No. 3 - Java Data Types

1. Write a Java program to count the letters, spaces, numbers, and other characters of an input string.
2. Implement a Java function that calculates the sum of digits for a given character array consisting of the digits '0' to '9'. The function should return the digit sum as a long value.
3. Find the smallest and largest element from the array.

Practical No. 4 - Methods and Constructors

1. Design a class `SortData` that contains the methods `asc()` and `desc()`.
2. Design a class that demonstrates the use of constructor and destructor.
3. Write a Java program to demonstrate the implementation of an abstract class.

Practical No. 5 - Inheritance

1. Write a Java program to implement single-level inheritance.
2. Write a Java program to implement method overriding.
3. Write a Java program to implement multiple inheritance.

Practical No. 6 - Packages and Arrays

1. Create a package, add the necessary classes, and import the package in a Java class.
2. Write a Java program to add two matrices and print the resultant matrix.
3. Write a Java program for multiplying two matrices and print the product for the same.

Practical No. 7 - Vectors and Multithreading

1. Write a Java program to implement vectors.
2. Write a Java program to implement thread life cycle.
3. Write a Java program to implement multithreading.

Practical No. 8 - File Handling

1. Write a Java program to open a file and display the contents in the console window.
2. Write a Java program to copy the contents from one file to another file.
3. Write a Java program to read the student data from the user and store it in the file.

Practical No. 9 - GUI and Exception Handling

1. Design an AWT program to print the factorial for an input value.
2. Design an AWT program to perform various string operations like reverse string, string concatenation, etc.
3. Write a Java program to implement exception handling.

Practical No. 10 - GUI Programming

1. Design an AWT application that contains the interface to add student information and display the same.
2. Design a calculator based on AWT application.
3. Design an AWT application to generate result marks sheet.

Reference Books:

1. SCJP Sun Certified Programmer by Kathy Sierra and Bert Bates
2. Complete Reference by TMH.
3. Java SE 8 for Programmers (3rd Edition) (Deitel Developer Series) by Paul Deitel and Harvey Deitel
4. www.tutorialspoint.com/java/
5. www.javatpoint.com/java-tutorial

Scheme of EoSE:

- 1) Exercise(s): 45
- 2) Viva Voce: 20
- 3) Evaluation of record book: 15

BCA III SEMESTER

Code of the course	: BCA6007T
Title of the course	: Operating System
Level of the Course	: NHEQF Level 5.0
Credit of the Course	: 4
Type of the Course	: Discipline Centric Core Course (DCC) in computer science
Delivery Type of the Course	: Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.
Prerequisite	: None
Objectives of the Course	: To provide basic knowledge of Operating System. Teach various features that help students to use and learn Unix OS.
Learning Outcomes	:
	1. To understand fundamental knowledge of Operating System.
	2. To learn various Linux commands and write shell programming.

UNIT-I

Operating System Overview: Introduction to Operating System, Functions of Operating System, Types of Operating system, User interface (CUI, GUI), Operating System Services, Multi-Tasking, Multi Programming, Multiprocessing. Additional Concept: Buffering, Spooling.

(12 Lectures)

UNIT-II

Process Management: Process, Process States, Process Control Block (PCB) Scheduling: Types of Schedulers. Measuring Performance of Computer System: Throughput, Turnaround time, Response time, Waiting Time, CPU burst. Scheduling Algorithms – FCFS, SJF, Priority & Round Robin (RR) Scheduling. Mutual exclusion, Deadlocks.

(12 Lectures)

UNIT-III

Memory Management: Static Memory Allocation, Dynamic Memory Allocation, Segmentation, Virtual Memory – Paging, Demand Paging, Page Replacement, Fragmentation & Defragmentation, Cache memory

(12 Lectures)

UNIT-IV

Linux Overview: Features of Linux, Basic commands of Linux, System structure, Shell and its feature, Types of shell, Kernel, Architecture of Linux

(12 Lectures)

UNIT-V

Shell Programming: Linux filesystem, Editors of Unix: (VI), Environment and User define variable, Argument processing, Shell interpretation at prompt, Arithmetic Expression evaluation.

(12 Lectures)

Text and References Book:

1. Silberschatz & Galvin: Operating System Concept, Wiley, Sixth Edition
2. Milan Malenkov's: Operating Systems, Tata McGraw-Hill, Second Edition.
3. William Stallings: Operating Systems, PHI, Fourth Edition
4. Yashavant Kanetkar: UNIX Shell Programming, BPB.

Suggested E-resources:

1. <https://www.tutorialspoint.com/unix/unix-useful-commands.html>
2. http://mally.stanford.edu/~sr/computing/basic_unix.html

BCA III SEMESTER

Code of the course	: BCA6007P
Title of the course	: Operating System Lab
Level of the Course	: NHEQF Level 5.0
Credit of the Course	: 2
Type of the Course	: Discipline Centric Core Course (DCC) in computer science
Delivery Type of the Course	: Practical 60. The student will perform the experiments and submit the record of observations after getting the results.
Prerequisite	: Basics knowledge of Operating System concepts of Operating System lab

Course Objective :

The course aims to provide basic knowledge of Operating System. This course intends to teach various features that will help the students to use and learn the working of Unix Operating System.

Learning Outcomes:

1. To understand fundamental knowledge of Operating System.
2. To learn various Linux commands and write shell programming.

Syllabus: Concepts of I/O programming, page removal algorithms & analysis, simulation of a traffic controller of a process scheduler, paging, segmentation, algorithms for "N" process Synchronization (critical section, semaphores). Algorithms for deadlock Detection & recovery, producer-consumer algorithm, UNIX shell programming. Familiarization with UNIX system calls for process management and inter-process communication, Experiments on process scheduling and other operating system tasks through simulation/implementation under a simulated environment (like Nachos).

List of Experiments:

1. Basic UNIX commands.
2. Write a shell script for arithmetic addition between two numbers.
3. Write a shell script for arithmetic Subtraction between two numbers.
4. Write a shell script for arithmetic Multiplication between two numbers.
5. Write a shell script for arithmetic Division between two numbers.
6. Write a shell script for Biggest of two Numbers.
7. Write a shell script for Biggest among Three Numbers.
8. Write a shell script for Implementation of Combination.
9. Write a shell script for Compound Interest Computation.
10. Write a shell script for Counting up to any Required Number.
11. Write a shell script for Sum of Digits of any 5-digit Number.
12. Write a shell script for Find out Whether a Number is Even or Odd?
13. Write a shell script for Fibonacci Series Display.
14. Write a shell script for Checking Palindrome.
15. Write a shell script for Simple Interest Calculation.
16. Write a shell script for swapping two Numbers Using Two Variables.
17. WAP to implement FCFS CPU scheduling algorithm with same arrival time using shell script. Also write the algorithm.
18. WAP to implement SJF CPU scheduling algorithm using shell script. Also write the algorithm.
19. WAP to implement Round Robin CPU scheduling algorithm using shell script also write the algorithm.

Scheme of EoSE:

- 1) Exercise(s): 45 2) Viva Voce: 20 3) Evaluation of record book: 15

BCA III SEMESTER

Code of the course : BCA6008T
Title of the course : Accounting and Financial Management
Level of the Course : NHEQF Level 5.0
Credit of the Course : 4
Type of the Course : Discipline Centric Core Course (DCC) in computer science
Delivery Type of the Course : Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving. Prerequisite: None

Objectives of the Course:

1. To become familiar with Tally ERP
2. This course is for individuals who are interested in learning about the core skills necessary for Accounting. This course is intended for individuals of all levels.

Learning Outcomes:

1. Identify the forms of business organizations and
2. explain the principal types of business activities.
3. Name the four primary financial statements and illustrate their main components.
4. List major assumptions and principles in financial reporting.

Unit-I

Introduction: History and Development of Accounting – Meaning Objectives and functions of Accounting- Book-keeping V/s Accounting – Users of accounting data – systems of book-keeping and accounting – branches of accounting – advantages and limitations of accounting. Accounting Concepts and conventions: Meaning need and classification, Accounting standards – meaning, need and classification of Indian Page 21 of 38 accounting standards. Accounting principles V/s Accounting standards.

(12 Lectures)

Unit-II

Financial Accounting Process: Classification of accounting transaction and accounts, rules of debit and credit as per Double Entry System. Journalisation and Ledger position Preparation of different subsidiary books: Purchase Day Book Sales Day Book, Purchase Returns Day Books, Sales Returns Day Book, Cash Book. Bank Reconciliation Statement: Meaning, Need, Definition, preparation of BRS.

(12 Lectures)

Unit- III

Accounting for bill of exchange: Meaning, Need, Definition, Parties to Bill of Exchange, Types of Bills. Accounts Procedure: Honour of the Bill, Dishonour of the Bill, Endorsement, Discounting, Renewal, Bills for collection, Retirement of the Bill, Accommodation Bills, Bill Receivable Book and Payable Book. Preparation of Trial Balance: Rectification of errors and journal Proper.

(12 Lectures)

Unit-IV

Preparation of Final accounts: Meaning, need and classification, Preparation of Manufacturing, Trading, Profit and loss account and Balance-Sheet of sale – traders and partnership firms.

(12 Lectures)

Unit V

Accounting Package like Tally

(12 Lectures)

Text and Reference Book

1. S. Ramesh, B.S. Chandrashekar, a Text Book of Accountancy.
2. V.A. Patil and J.S. Korihalli, Book–Keeping and Accounting, (R. Chand and Co. Delhi).
3. R.S. Singhal, Principles of Accountancy, Nageen Prakash pvt. Ltd, Meerut.
4. B.S. Raman, Accountancy, (United Publishers, Mangalore)

Suggested E-resources:

1. The 8 Important Steps in the Accounting Cycle (investopedia.com)
2. Financial Accounting Notes.pdf (josephcollege.ac.in)
3. Introduction to Final Accounts and Accounting Treatment (taxmann.com)

BCA III SEMESTER

Code of the course	: BCA6008P
Title of the course	: Accounting and Financial Management Lab
Level of the Course	: NHEQF Level 5.0 Credit of the Course: 2
Type of the Course	: Discipline Centric Core Course (DCC) in computer science
Delivery Type of the Course	: Practical 60. The student will perform the experiments and submit the record of observations after getting the results.
Prerequisite	: None

Objectives of the Course:

1. To become familiar with Tally ERP
2. This course is for individuals who are interested in learning about the core skills necessary for Accounting. This course is intended for individuals of all levels.

Learning Outcomes:

1. Identify the forms of business organizations and explain the principal types of business activities.
2. Name the four primary financial statements and illustrate their main components.
3. List major assumptions and principles in financial reporting.

LIST OF EXPERIMENTS:

Note: The lab instructor is expected to design problems based on the topics listed.

UNIT I

Basics Concepts of Accounts: - Types of Accounts, Golden Rules of Accounting, Accounting Principles, Concepts and Conventions, Double Entry System of Book Keeping, Mode of Accounting, Financial Statements, Transactions, Recording Transactions Fundamentals of Tally. ERP 9: - Getting Functional with Tally. ERP 9, Creation/Setting up of Company in Tally. ERP 9, Journal Entries Accounting Masters in Tally. ERP 9: - F11: Features, F12: Configurations. Setting up Account Heads

(12 Lectures)

UNIT II

Inventory in Tally. ERP 9: - Stock Groups, Stock Categories, Godowns / Locations, Units of Measure Stock Items, Creating Inventory Masters for National Traders Voucher Entry in Tally. ERP 9: - Accounting Vouchers, Inventory Vouchers, Invoicing

(12 Lectures)

UNIT III

Advanced Accounting in Tally, ERP 9: - Bill-wise Details, Cost Centres and Cost Categories, Voucher Class and Cost Centre Class, Bank Reconciliation, Interest Calculations, Budgets & Controls, Scenario Management

(12 Lectures)

UNIT IV

Accounting Voucher: - Contra Vouchers, Payment Vouchers, Receipt Vouchers, Sales Vouchers, Purchase Vouchers, Journal Vouchers

Technological Advantages of Tally. ERP 9: -Tally Vault, Debit and Credit Notes, Debit Note Returns with Examples, Credit Note Returns with Examples, Security Control, Tally Audit, Backup and Restore, Split Company Data

(12 Lectures)

UNIT V

Goods and Services Tax (GST): -About Goods and Service (GST), Activating Tally in GST, Setting Up GST (Company Level, Ledger Level or Inventory, Level), GST Taxes & Invoices, Understanding SGST, CGST & IGST, Creating GST Masters in Tally

(12 Lectures)

Text and Reference Book:

1. GST Accounting with Tally. ERP 9 By Asok K Nadhani Tally Essential Level 1 By Tally Education Pvt. Ltd.
2. Tally Essential Level 2 By Tally Education Pvt. Ltd.

Suggested E-resources:

1. The 8 Important Steps in the Accounting Cycle (investopedia.com)
2. FinancialAccountingNotes.pdf (josephcollege.ac.in)
3. Introduction to Final Accounts and Accounting Treatment (taxmann.com)

Scheme of EoSE: -

1) Exercise(s): 45

2) Viva Voce: 20

3) Evaluation of record book: 15

BCA IV SEMESTER

Code of the course : BCA6009T
Title of the course : System analysis and design
Level of the Course : NHEQF Level 5.0
Credit of the Course : 4
Type of the Course : Discipline Centric Compulsory (DCC) in computer science
Delivery Type of the Course : Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.

Prerequisites : None

Objectives of the Course :

1. Define the system development life cycle.
2. Conducts research on existing systems.
3. Develop plans for the new system.
4. Make the feasibility study about the system.
5. Scheduling with using GANTT and PERT techniques.
6. Evaluates the economic self-sufficiency whether to install the system.
7. carry out the system analysis.
8. Identifies problems in the system.
9. Determine the cause of the problem in the system.

Course Learning Outcomes:

1. Knowledge and understanding - Understand the principles and tools of systems analysis and design
- Understand the application of computing in different context - Understand the professional and ethical responsibilities of practicing the computer professional including understanding the need for quality
2. Cognitive skills (thinking and analysis). - Solve a wide range of problems related to the analysis, design and construction of information systems - Analysis and Design of systems of small sizes
3. Communication skills (personal and academic). - Be able to present projects
4. Practical and subject specific skills (Transferable Skills). - Plan and undertake a major individual project, prepare and deliver coherent and structured verbal and written technical reports

UNIT-I

Introduction: System Concept and the need for system approach, Definition of system and system analysis, Factoring into subsystems, Black box system, Introduction to the basic elements of the system, Different types and behavior of the system.

(12 Lectures)

UNIT-II

The System Development Life Cycle and System Analyst: Source and inspiration of a new system development, Recognition and need, Linear approach and prototype approach, Different phases in SDLC, Role of System Analyst.

(12 Lectures)

UNIT-III

System Analysis: Importance of planning and control, Information Gathering: Various Methods, Tools of Structured Analysis: DFD, Decision Tree, Structured English, Decision Tables, Data Dictionary, Feasibility study. System Design: The Process of Design: Logical and Physical design, Methodologies: Structured, Form-Driven, IPO Charts etc., Input Output Form Design, File Organization: Sequential Indexed, inverted list, Database Design, Logical and Physical View of Data.

(12 Lectures)

UNIT-IV

System Implementation: Need of Testing, Test Plan, Quality Assurance, Trends in Testing, Audit Trail, Post Implementation Review, Project Scheduling, Selection of Hardware and Software

(12 Lectures)

UNIT-V

Security and Recovery in System Development: System Security: Definition, Threats to system security, Control measures, Disaster/ Recovery Planning, Ethics in System Development. Case Study.

(12 Lectures)

Text and References books

1. System Analysis and Design -E.M. Awad
2. System Analysis and Design - Dennis Wixom
3. Systems Analysis and Design: Kenneth E. Kendall and Julie E. Kendall Publisher: Prentice Hall PTR, 5th Edition, 2001

Suggested E-resources.

1. <https://www.geeksforgeeks.org/complete-roadmap-to-learn-system-design/>
2. <https://nios.ac.in/media/documents/vocational/cca/cca1.pdf>
3. <https://www.educative.io/blog/system-analysis-vs-system-design>
4. <https://study.com/academy/course/computer-science-302-system-analysis-design.html>

BCA IV SEMESTER

Code of the course : BCA6009P
Title of the course : System analysis and Design Lab
Level of the Course : NHEQF Level 5.0
Credit of the Course : 2
Type of the Course : Discipline Centric Compulsory (DCC) Course in computer science
Delivery Type of the Course : Practical 60. The student will perform the experiments and submit the record of observations after getting the results.
Prerequisites : None

Objectives of the Course:

To prepare the students to develop the skills necessary to develop different diagrams. To make the students aware of the importance of software engineering principles in designing software projects

Course Learning Outcomes: On completion of the course the student can able to develop

1. Standard SRS document
2. Different Diagrams for given software.

List of experiments:

1. Develop an IEEE standard SRS document.
2. Discuss the tool to draw different types of diagrams throughout the analysis & design.
3. Develop Data Flow Diagrams
4. Identify Use cases and develop Use case model
5. Develop Activity Diagram
6. Develop State Diagram
7. Develop Sequence Diagram
8. Develop Collaboration Diagram
9. Develop Entity Relationship Diagram
10. Develop Use cases, Sequence diagram and Activity Diagram for Event management system
11. Develop Use cases, Sequence diagram and Activity Diagram for Payroll management system
12. Develop DFD, ERD and Use cases, for Student Feedback System
13. Develop DFD, ERD and Use cases, for Inventory Management System
14. Develop DFD, ERD, Use cases, Sequence diagram and Activity Diagram for Attendance Management

Text and References books

1. System Analysis and Design -E.M. Awad
2. System Analysis and Design - Dennis Wixom
3. Systems Analysis & Design: Kenneth E. Kendall Julie E. Kendall: Prentice Hall PTR, 5th Edition, 2001

Suggested E-resources

1. <https://www.studocu.com/in/document/lords-universal-college/computer-applications/yash-sad-lab-manual-hi/37681173>
2. <https://elearn.daffodilvarsity.edu.bd/course/view.php?id=6124> 3.
3. <http://personal.ee.surrey.ac.uk/Personal/J.Clark/teaching/sad/labs/COM1012%20Lab%20Sessions%20-%201.0.0rc1.htm>

Scheme of EoSE:

1) Exercise(s): 45

2) Viva Voce: 20

3) Evaluation of record book: 15

BCA IV SEMESTER

Code of the course	: BCA6010T
Title of the course	: C # with Dot net framework
Level of the Course	: NHEQF Level 5.0
Credit of the Course	: 4
Type of the Course	: Discipline Centric Compulsory (DCC) Course in computer science
Delivery Type of the Course	: Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.
Prerequisites	: None

Objectives of the Course :

1. Introduce to .Net IDE Component Framework.
2. Programming concepts in .Net Framework.
3. Creating website using ASP.Net Controls

Learning outcomes :

1. The student will be able to Acquire the knowledge of the structure and model of the programming language C #
2. Understand the use of programming language C # for various programming technologies
3. Evaluate user requirements for software functionality required to decide whether the programming language C # can meet user requirements
4. Develop variety of software in C #

Unit I

The .NET Framework: Introduction, Common Language Runtime, Common Type System, Common Language specification, The Base Class Library, The .Net class library Intermediate language, Just-in time Compilation, Garbage Collection, Application Installation and Assemblies, Web services, Unified classes.
(12 Lectures)

Unit II

C# Basics: Introduction, Data Types, Identifiers, Variables and constants, C# statements, Object Oriented Concept, Object and Classes, Arrays and Strings, System collections, Delegates and Events, Indexes, Attributes, versioning.
(12 Lectures)

Unit III

C# Using Libraries: Namespace- System, Input Output, Multi-Threading, Networking and Sockets, Data Handling, Windows Forms, C# in web application, Error Handling
(12 Lectures)

Unit IV

Advanced Features Using C#: Web services, Windows services, messaging, Reflection, COM and C#, Localization.
(12 Lectures)

Unit V

Advanced Features Using C#: Distributed Application in C#, XML and C#, Unsafe Mode, Graphical Device Interface with C#, CASE Study (Messenger Application)
(12 Lectures)

Text and References Book:

- (1) Jeffrey Richter, “Applied Microsoft .NET Framework Programming”, (Microsoft)
- (2) Fergal Grimes, “Microsoft .Net for Programmers”, (SPD)
- (3) Balagurusamy, “Programming with C# “, TMH Wiley,” Beginning Visual C# 2008”, Wrox

Suggested E-resources:

1. [C# Tutorial \(C Sharp\) \(w3schools.com\)](http://w3schools.com)
2. [Introduction to .NET Framework - Jvatpoint](http://jvatpoint.com)
3. [Advanced Features In C# \(c-sharpcorner.com\)](http://c-sharpcorner.com)

BCA IV SEMESTER

Code of the course : BCA6010P
Title of the course : C # with Dot net framework Lab
Level of the Course : NHEQF Level 5.0
Credit of the Course : 2
Type of the Course : Discipline Centric Compulsory (DCC) in computer science
Delivery Type of the Course : Practical 60. The student will perform the experiments and submit the record of observations after getting the results.
Prerequisites : None
Objectives of the Course :

1. Introduce to .Net IDE Component Framework.
2. Programming concepts in .Net Framework.
3. Creating website using ASP.Net Controls

Learning outcomes:

1. The student will be able to Acquire the knowledge of the structure and model of the programming language C #
2. Understand the use of programming language C # for various programming technologies
3. Evaluate user requirements for software functionality required to decide whether the programming language C # can meet user requirements
4. Develop variety of software in C #

List of Experiment

1. Getting started with technology and writing basic programs like Hello World in C#.
2. Programs to demonstrate the usage of classes and objects.
3. Program in C# to demonstrate the usage of Decision Making, Looping and Branching.
4. Program in C# to demonstrate the concepts of Array, Multidimensional Array, and jagged Array.
5. Program in C# to demonstrate boxing and unboxing.
6. Program in C# to demonstrate single level inheritance and multilevel inheritance.
7. Program in C# to demonstrate interfaces.
8. Program in C# to demonstrate virtual and override keyword, operator overloading, abstract classes and methods.

Text and References Book:

- (1) Jeffrey Richter, "Applied Microsoft .NET Framework Programming", (Microsoft)
- (2) Fergal Grimes, "Microsoft .Net for Programmers", (SPD)
- (3) Balagurusamy, "Programming with C# ", TMH Wiley, " Beginning Visual C# 2008", Wrox

Suggested E-resources:

1. [C# Tutorial \(C Sha/rp\) \(w3schools.com\)](#)
2. [Introduction to .NET Framework - Javatpoint](#)
3. [Advanced Features In C# \(c-sharpcorner.com\)](#)

Scheme of EoSE: -

- (1) Exercise(s): 45 (2) Viva Voce: 20 (3) Evaluation of record book: 15

BCA IV SEMESTER

Code of the course : BCA6011T
Title of the course : Computer Communication and Networks
Level of the Course : NHEQF Level 5.0
Credit of the Course : 4
Type of the Course : Discipline-Centric Compulsory (DCC) in computer science
Prerequisites : None
Delivery Type of the Course : Lecture, 40 + 20 = 60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem-solving.
Objectives of the Course :

1. To understand basics of Cryptography and Network Security.
2. To be able to secure a message over insecure channel by various means.
3. (3) To learn about how to maintain the Confidentiality, Integrity and Availability of a data.
4. To understand various protocols for network security to protect against the threats in the networks.

Learning Outcomes:

1. Provide security of the data over the network.
2. Do research in the emerging areas of cryptography and network security.
3. Implement various networking protocols.
4. Protect any network from the threats in the world.

UNIT-I

Security and Cryptographical algorithm: Need for security, principle of security, types of attacks. Cryptographic techniques: cryptography terminology, substitution techniques, transposition techniques, Symmetric and asymmetric key algorithm, possible types of attack, key range, steganography. Symmetric vs asymmetric, algorithm types and modes, DES, double and triple DES, AES, and comparison of various cryptographical algorithms and requirement of good cryptographical algorithm.

(12 Lectures)

UNIT-II

Asymmetric cryptographical algorithm and Message Authentication: Public key cryptography principles and algorithms, RSA algorithm, Diffie-Hellman key exchange. One-way hash functions, message digest, MD5, SHA1, message authentication code, Digital envelope, and Digital signatures.

(12 Lectures)

UNIT-III

Network Management: Management Standards and Models, configuration management, configuration database and reports, fault management, identification and isolation, protecting sensitive information, host and user authentication, structure of management information, Standard management information base, SNMPv1 protocol, accounting management, performance management, network usage, matrices, and quotas.

(12 Lectures)

UNIT-IV

Network security: Overview of IPV4: OSI model, maximum transfer unit, IP, TCP, UDP, ICMP, ARP, RARP, and DNS, ping, traceroute. Network attacks: Buffer overflow, IP scheduling, TCP session hijacking, sequence guessing. Network scanning: ICMP, TCP sweeps, basic port scans. Denial of service attacks: SYN flood, teardrop attacks, land, smurf attacks. Visual and private network topology: tunneling, IPSEC. Traffic protocols: authentication headers, ESP, internet key exchange, security association, PPTP, L2TP.

(12 Lectures)

UNIT-V

Firewalls: Firewall characteristics & design principles, types of firewalls, packet filtering router, application-level gateway or proxy, content filters, bastion host. Firewall architectures: dual homed host, screening router, screened host, screened subnet. Firewall logs.

Instruction detection system: component of an IDS, placement of IDS components, types of IDS: network-based IDS, file integrity checkers, host-based IDS, IDS evaluation parameters.

(12 Lectures)

Recommended Books

1. Network Security, Private Communication in a public world, Kaufman, C.
2. Perlman, R., and Speciner, M., Prentice Hall.
3. Cryptography and Network Security: Principles and Practice, Stallings, W. Prentice Hall PTR.
4. Fundamentals of Computer Security, Pieprzyk Josef, Springer-Verlag,
5. W. Mao, "Modern Cryptography – Theory and Practice", Pearson Education.

Suggested E-resources

1. [<https://ecomputernotes.com/computernetworkingnotes/security/requirements-of-network-security>]
2. <http://nptel.ac.in/courses/106105031/lecture> by Dr. Debdeep Mukhopadhyay, IIT Kharagpur
3. [<https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-033-computer-system-engineering-spring-2009/video-lectures>] by Prof. Robert Morris and Prof. Samuel Madden MIT.

BCA IV SEMESTER

Code of the course : BCA6011P
Title of the course : Computer Communication and Networks-lab
Level of the Course : NHEQF Level 5.0
Credit of the Course : 2
Type of the Course : Discipline-Centric Compulsory (DCC) Course in computer science
Delivery Type of the Course : Practical 60. The student will perform the experiments and submit the record of observations after getting the results.

Prerequisites : None

Objectives of the Course :

1. To understand the working principle of various communication protocols.
2. To analyze the various routing algorithms.
3. To know the concept of data transfer between nodes

Course Learning Outcomes:

1. Identify and use various networking components
2. Understand different transmission media and design cables for establishing a network
3. Implement any topology using network devices
4. Analyze performance of various communication protocols.
5. Compare routing algorithms
6. Understand the TCP/IP configuration for Windows and Linux
7. Implement device sharing on network
8. Learn the major software and hardware technologies used on computer networks

List of Experiments:

1. Study of different types of Network cables and Practically implement the cross-wired cable and straight-through cable using clamping tool.
2. Study of Network Devices in Detail.
3. Study of network IP.
4. Connect the computers in Local Area Network.
5. Study of basic network command and Network configuration commands.
6. Socket Program for Echo/Talk commands.
7. Configure a Network topology using packet tracer software.
8. Write a code simulating PING and TRACEROUTE commands.
9. Basic Ethernet LANs and Switches.
10. Ethernet Design, VLANs & Troubleshooting.
11. Advanced Ethernet VLANs, STP, and Trunking.
12. IPv4 Subnetting.
13. Implementing IPv4 on routers.
14. Access Lists and NAT.
15. IPv6 Subnetting.
16. SDN and Cloud.
17. Implementation of STOP & WAIT protocol and sliding window protocol.
18. Write a program to implement subnetting and find the subnet masks.
19. TCP/IP protocol architecture.
20. User Datagram Protocol (UDP).
21. Multicasting.
22. Transmission Control Protocol (TCP).
23. Standard Internet services, and protocol usage by common Internet applications.
24. Sockets programming.

25. Client/server, peer-to-peer, Internet addressing, TCP sockets, UDP sockets.

Text and Reference Books

1. Data & Communications, William Stallings.
2. Computer Networks, A.S. Tanenbaum.
3. Data Communications and Networking, Behrouz A. Forouzan, McGraw-Hill Higher Ed.
4. Computer Networking: A Top-Down Approach, James F. Kurose, Keith W. Ross, Pearson Education.

Suggested E-resources:

1. [<https://onl.kmi.open.ac.uk/resources/>]
2. [<https://www.cisco.com/site/us/en/products/networking/wireless/index.html>]
3. [https://www.youtube.com/watch?v=0pMm_QxCg3I]
4. [https://www.cisco.com/c/en_in/solutions/enterprise-networks/what-is-computer-networking.html#~types-of-networks]

Scheme of EoSE :

- (i) Exercise(s):45 (ii) Viva Voce: 20 (iii) Evaluation of record book: 15

BCA IV SEMESTER

Code of the course : SEI6330T
Title of the course : Basic Mathematics
Level of the Course : NHEQF Level 5.0
Credit of the Course : 2
Type of the Course : Skill Enhancement Course (SEC) Course in computer science
Delivery Type of the Course : 30 Lectures

Prerequisites : Candidates should have completed their 10+2 or equivalent examination from a recognized board with a minimum of 40% aggregate marks in any stream.

Objectives of the Course:

1. The course aims to strengthen the conceptual knowledge of Mathematics learned at the School level.
2. Lay the foundation for further learning of the subject through the first course on Basic Mathematics.

Learning Outcomes:

1. Algebraic Expressions
2. Linear Equations
3. Quadratic Equation of lines
4. Radical expressions and complex numbers
5. Conic Sections

Unit I

Algebraic Expressions: Term and degree

Evaluating Algebraic Expressions: Addition and Subtraction of Algebraic Expressions, Multiplication & Division of Algebraic Expressions, Rational Expressions, Operations with Rational Expressions

Factoring: Difference of squares, quadratic trinomials, splitting middle term

(6 Lectures)

Unit II

Linear Equations:

Translating algebraic expressions

Solving linear equations Addition property

Solving linear equations: Multiplication property

Combining rules

Inequalities

Solving linear inequalities

Graphing Linear Equations

The Cartesian coordinate system

The graph of a line equation

Solve Linear equations in two variables by graph

(6 Lectures)

Unit III

Systems of Linear Equations

Solution of Systems of equations in two variables (Addition/Elimination Method)

Quadratic equations

Solution by Special methods: by square root, by Factorization of roots, Completing the square, The quadratic formula. Nature of roots, Sum and product of roots

(6 Lectures)

Unit IV

Radical expressions and complex numbers

Introduction to roots and radicals

Simplifying radical expressions [No variables]

Operations with radical expressions, rationalizing binomials denominators

Complex numbers [Addition and Subtraction]

Equation of lines

Slope of a line, Parallel & perpendicular line, slope intercept form of equation of the line, slope point form, two-point form, intercept form

(6 Lectures)

Unit V

Conic Sections: General quadratic equation, conic sections, circles, parabolas, ellipses, hyperbolas

System of Real Numbers: Natural, whole, integer, rationales, irrationals, graphical representation of real numbers

(6 Lectures)

Text and Reference Books

1. Discrete Mathematics, Schaum's Outlines
2. Differential Calculus, Shanti Narayan, P.K. Mittal
3. Integral Calculus, Shanti Narayan, P.K. Mittal
4. Elementary Calculus, Gokhroo & Bhargav.
5. Business Mathematics, Quaji Zameeruddin, V.K. Khanna, S.K. Bhambri
6. Comprehensive Mathematics Class XII Part-A, Parmanand Gupta

BCA IV SEMESTER

Code of the course	: SEI6331T
Title of the course	: Fundamental Web Development
Level of the Course	: NHEQF Level 5.0
Credit of the Course	: 2
Type of the Course	: Skill Enhancement Course (SEC) Course in computer science
Delivery Type of the Course	: 30 Lectures
Prerequisites	: None

Objectives of the Course:

1. Giving the students the insights of Internet programming
2. To design and implement complete applications over the web.

Learning Outcomes:

1. Analyze a web page and identify its elements and attributes.
2. Create web pages using XHTML and Cascading Style Sheets.
3. Build dynamic web pages using JavaScript (Client-side programming).
4. Create XML documents and Schemas.
5. Build interactive web applications using AJAX.

UNIT-I

HTML: Introduction, Basic structure of an HTML document, Mark up Tags, Heading-Paragraphs, Line Breaks, HTML Tags, Elements of HTML, Working with Text, Working with Lists, Tables and Frames, Working with Hyperlinks, Images and Multimedia, Working with Forms and controls.

(6 Lectures)

UNIT-II

CSS: Creating Style Sheet, CSS Properties, CSS Styling (Background, Text Format, Controlling Fonts), Working with block elements and objects, Working with Lists and Tables, CSS Id and Class, Box Model (Introduction, Border properties, Padding Properties, Margin properties), Introduction to Web Publishing or Hosting.

(6 Lectures)

UNIT-III

JavaScript: Variable, Naming Rules, Data Types, Expressions and Operators, Flow Control, Objects and Arrays Functions and Methods Pattern Matching with Regular Expressions Managing Web Page Styles using JavaScript and CSS Introduction to Ajax.

(6 Lectures)

UNIT-IV

Understanding XML: SGML, XML, XML and HTML, Modeling XML Data, Styling XML with XSL, XHTML Request and Response Objects, Cookies, Working with Data OLEDB connection class, command class, transaction class, data adaptor class, data set class.

(6 Lectures)

UNIT-V

Current trends and Issues: Email, Application Issues, Working with IIS and page Directives, Error handling. Security - Authentication, IP Address, Secure by SSL & Client Certificates.

(6 Lectures)

Text Books and References:

1. Harvey M. Dietel, Paul Dietel & Tem R. Nieto, "Internet & World Wide Web How to Program", Pearson, 2011.
2. Ivan Bayross. "Web-enabled commercial application development using HTML, DHTML, JavaScript, PERL-CGI", BPB Publications, 2010
3. Mobile Communications, Jochen Schiller, Addison Wesley Pearson education.

Suggested E-resources:

1. <https://startuptalky.com/learn-web-development-online-resources>
2. <https://www.geeksforgeeks.org/web-technology>
3. <https://study.com/academy/lesson/what-is-web-technology-definition-trends.html>
4. <https://www.javatpoint.com/web-application>
5. <https://www.techtarget.com/whatis/definition/XML-Extensible-Markup-Language>

BCA IV SEMESTER

Code of the course	: SEI6332T
Title of the course	: Business Organisation and Management
Level of the Course	: NHEQF Level 5.0
Credit of the Course	: 2
Type of the Course	: Skill Enhancement Course (SEC) Course in computer science
Delivery Type of the Course	: 30 Lectures
Prerequisites	: None

Objectives of the Course :

To acquaint the students with the concept of business organization and applicable tax regime. This syllabus lays down the foundation for actual functioning of an organization, concepts of general management and its implementation in organization

Course Learning Outcomes:

1. Students will be able to understand the concepts and form of Business organizations
2. Students will be able to study contributions of Taylor, Henry Fayol and understand various systems and approaches to management.
3. To demonstrate the planning and decision making approaches by using appropriate strategies to accomplish a given job.
4. Students will be able to understand the need and importance of organization process & structure, span of control and delegation of authority.
5. The students would be able to have strong foundation and practical knowledge about basic concepts of business organisation.

UNIT-I

Business and Management: Business Meaning and Contents, Business as a system, forms of business organization: Sole Trader, Partnership firm, HUF, company, LLP), Formation & choice of business organization. Business Environment: It's Nature, types and effect on business.

(6 lectures)

UNIT-II

Management Definition, Concept and Nature, Management Process, Basic function of Management, Management Level, Role of Manager, Management Principles (Henry Fayol's principle of management, Taylor's Scientific Management). Planning: Definition - Importance - Steps in planning – limitations - Types of Plans

(6 lectures)

UNIT-III

Organizational Behavior: Need of Understanding human behavior in organization, Challenges and Opportunities for OB. Management by Objective (MBO), Decision making: Definition – Process – Types of decisions: – Programmed and Non-programmed decisions – Strategic and routine decisions- major and minor decisions – Individual and group decisions. Decision Making process and Models.

(6 lectures)

UNIT-IV

Managing Personnel: HRM- Meaning and Functions, Man Power Planning, Job Analysis and Design, Training, Career Planning & Development. Conflict Management, Strategies & Policies. Motivation Theories & Practices, Leadership Concept theories & Style, Compensation Management.

(6 lectures)

UNIT-V

Marketing Management and Finance: Basic Concepts of Marketing, Nature & Scope of Marketing, Sales Promotion, Product Life Cycle, Marketing Information System (MIS) and Marketing Research. Main Sources of Finance, Concept of Fixed & Working Capital, Introduction of Tax – Income Tax, GST, Basic Concept of Invoice & Quotations.

(6 lectures)

Suggested Books and References

1. B.P. Singh & T.N. Chhabra, “Business Organization and Management Functions”, Dhanpat Rai & Co. 2000.
2. P.C Tripathi
3. & P.N. Reddy, “Principles of Management”, Tata McGraw Hill Publishing Company New Delhi.
4. L.M. Prasad, “Principles and Practices of Management”.
5. Stephen P. Robbins, “Organizational Behaviour”, (8th Ed.) Prentice Hall of India.
6. K. Aswathappa, “Human Resource Management”, Tata McGraw Hill Publishing Company New Delhi.
7. Philip Kotler, “Marketing Management”, (9th Ed.) Prentice Hall of India.
8. Ramaswamy.V.S. and Namakumari.S. “Marketing Management: Planning, Control.” New Delhi, MacMillan. 1990.
9. Dr.S.N. Maheshwari, “Financial Management – Principles and Practices” (6th revised Ed.) S.Chand & Sons.

Suggested E-resources:

1. http://epgp.inflibnet.ac.in/epgpdata/uploads/epgp_content/food_technology/food_business_management/07.planning_and_decision_making/et/2733_et_m7.pdf
2. <https://study.com/academy/lesson/what-is-conflict-management-definition-styles-strategies.html>
3. <https://www.mindtools.com/asjiu77/henri-fayols-principles-of-management>
4. <https://www.managementstudyguide.com/management-by-objectives.html>
5. <https://www.indeed.com>

Scheme of EoSE :

(i) Exercise(s):45 (ii) Viva Voce: 20 (iii) Evaluation of record book: 15

BCA IV SEMESTER

Course Code : BCA7100T
Title of the Course : Introduction to Data Mining
Level of the Course : NHEQF Level 5.5
Credit of the Course : 4
Type of the Course : Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course : Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.

Prerequisite:

1. A course on "Database Management Systems"
2. Knowledge of probability and statistics

Objectives of the Course:

1. It presents methods for mining frequent patterns, associations, and correlations.
2. It then describes methods for data classification and prediction, and data-clustering approaches.
3. It covers mining various types of data stores such as spatial, textual, multimedia, streams

Learning Outcomes:

1. Ability to understand the types of the data to be mined and present a general classification of tasks and primitives to integrate a data mining system.
2. Apply preprocessing methods for any given raw data.
3. Extract interesting patterns from large amounts of data.
4. Discover the role played by data mining in various fields.
5. Choose and employ suitable data mining algorithms to build analytical applications
6. Evaluate the accuracy of supervised and unsupervised models and algorithms

UNIT-I

Introduction to data mining Data Mining overview, Data Warehouse and OLAP Technology, Data Warehouse Architecture, Steps for the Design and Construction of Data Warehouses, A Three-Tier Data Warehouse Architecture, OLAP, OLAP queries, metadata repository

(12 Lectures)

UNIT-II

Data Preprocessing – Data Integration and Transformation, Data Reduction Data Mining Primitives: Defines a Data Mining Task, Task-Relevant Data, The Kind of Knowledge to be Mined, Various Architectures of Data mining Systems, Concept Description, Characterization and Comparison, Data Generalization and summarization

(12 Lectures)

UNIT-III

Mining Association Rules in Large Databases, Association Rule Mining, Market Basket Analysis: Mining A Road Map, The Apriori Algorithm: Finding Frequent Item sets Using Candidate Generation, Generating Association Rules from Frequent Itemsets, Improving the Efficiency of Apriori, Mining Frequent Itemsets without Candidate Generation, Multilevel Association Rules, Approaches to Mining Multilevel Association Rules, Mining Multidimensional Association Rules for Relational Database and Data Warehouses , Multidimensional Association Rules, Mining Quantitative Association Rules, Mining Distance-Based Association Rules, From Association Mining to Correlation Analysis

(12 Lectures)

UNIT-IV

Classification and Prediction: Introduction, Issues, Decision Tree Induction, Bayesian Classification, Classification based on Concepts from Association Rule Mining, Other Methods. Prediction, Introduction, Classifier Accuracy.

(12 Lectures)

UNIT-V

Cluster Analysis: Introduction, Types of Data in Cluster Analysis, Partitioning Methods, Hierarchical Methods, Density Based Methods , GRID Based Method, Model based Clustering Method.

(12 Lectures)

Text and Reference Books:

1. Data Mining Concepts and Techniques - Jiawei Han, Micheline Kamber, Morgan Kaufmann Publishers, Elsevier, 2 Edition, 2006.
2. Introduction to Data Mining, Pang-Ning Tan, Vipin Kumar, Michael Steinbach, Pearson Education
3. Data Mining Techniques, Arun K Pujari, 3rd Edition, Universities Press.
4. Data Warehousing Fundamentals, Pualraj Ponnaiah, Wiley Student Edition.
5. The Data Warehouse Life Cycle Toolkit - Ralph Kimball, Wiley Student Edition
6. Data Mining, Vikaram Pudi, P Radha Krishna, Oxford University.

Suggested E-resources:

1. <https://www.javatpoint.com/data-mining>
2. <https://www.geeksforgeeks.org/data-preprocessing-in-data-mining>
3. https://www.tutorialspoint.com/data_mining/data_mining_classification_and_prediction.htm

Scheme of EoSE:

- 1) Exercise(s): 45
- 2) Viva Voce: 20
- 3) Evaluation of record book: 15

BCA IV SEMESTER

Course Code : BCA7100P
Title of the Course : Introduction to Data Mining Lab
Level of the Course : NHEQF Level 5.5
Credit of the Course : 2
Type of the Course : Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course : Practical 60.
Prerequisite :

The knowledge of following subject is essential to understand the subject:

1. Understand the concepts of Data Warehousing and Data Mining Concepts.
2. Explain the methodologies used for analysis of data
3. Describe various techniques which enhance the data modeling.
4. Discuss and Compare various approaches with other techniques in data mining and data warehousing

Course Objectives:

Practical exposure on implementation of well-known data mining tasks. Exposure to real-life data sets for analysis and prediction. Learning performance evaluation of data mining algorithms in a supervised and an unsupervised setting. Handling a small data mining project for a given practical domain.

Learning Outcomes:

1. Ability to understand the types of the data to be mined and present a general classification of tasks and primitives to integrate a data mining system.
2. Apply preprocessing methods for any given raw data.
3. Extract interesting patterns from large amounts of data.
4. Discover the role played by data mining in various fields.
5. Choose and employ suitable data mining algorithms to build analytical applications.
6. Evaluate the accuracy of supervised and unsupervised models and algorithms

List of Experiments:

- 1) Fundamentals of Data Mining
- 2) Data Mining functionalities
- 3) Classification of data mining system
- 4) Pre-processing Techniques
- 5) APRIORI Algorithm
- 6) FP-Growth Algorithm
- 7) Spatial data mining
- 8) Web mining
- 9) Trends and applications of data mining
- 10) CASE STUDIES/ SMALL PROJECTS: Case study-1: Search queries on biomedical databases, such as PubMed, often return a large number of results, only a small subset of which is relevant to the user. Ranking and categorization, which can also be combined, have been proposed to alleviate this information overload problem. Results categorization for biomedical databases is the focus of this work. A natural way to organize biomedical citations is according to their MeSH annotations. First, the query results are organized into a navigation tree

List of Reference Experiments:

- 1) Introduction to Weka Explorer interface.
- 2) Introductory assignments of different data-set.
- 3) Assignment on the basis of nominal and numerical data set.
- 4) Practical assignment on the basis of classifications algorithm.

5) Practical assignment on the basis of Association algorithm

Text and Reference Books:

1. Data Mining Concepts and Techniques - Jiawei Han, Micheline Kamber, Morgan Kaufmann Publishers, Elsevier, 2 Edition, 2006.
2. Introduction to Data Mining, Pang-Ning Tan, Vipin Kumar, Michael Steinbach, Pearson Education
3. Data Mining Techniques, Arun K Pujari, 3rd Edition, Universities Press.
4. Data Warehousing Fundamentals, Pualraj Ponnaiah, Wiley Student Edition.
5. The Data Warehouse Life Cycle Toolkit - Ralph Kimball, Wiley Student Edition
6. Data Mining, Vikaram Pudi, P Radha Krishna, Oxford University.

Suggested E-resources:

- 1) <https://www.javatpoint.com/data-mining>
- 2) <https://www.geeksforgeeks.org/data-preprocessing-in-data-mining>
- 3) https://www.tutorialspoint.com/data_mining/data_mining_classification_and_prediction.htm

Scheme of EoSE:

- 1) Exercise(s): 45
- 2) Viva Voce: 20
- 3) Evaluation of record book: 15

BCA IV SEMESTER

Course Code : BCA7101T
Title of the Course : Python Programming
Level of the Course : NHEQF Level 5.5
Credit of the Course : 4
Type of the Course : Discipline Specific Elective (DSE) Course in Computer Science Delivery
Type of the Course: Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.
Prerequisite :
Experience with a high level language (C/C++, Java) is suggested. Prior knowledge of a scripting language (Perl, UNIX/Linux shells) and Object-Oriented concepts are helpful but not mandatory.

Objective of the course:

1. To learn how to design and program Python applications.
2. To learn how to use lists, tuples, and dictionaries in Python programs.
3. To learn how to identify Python object types.
4. To define the structure and components of a Python program.
5. To learn how to write loops and decision statements in Python.
6. To learn how to write functions and pass arguments in Python.
7. To learn how to build and package Python modules for reusability.
8. To learn how to read and write files in Python.
9. To learn how to design object-oriented programs with Python classes.
10. To learn how to use exception handling in Python applications for error handling.

Learning Outcomes:

1. Interpret the fundamental Python syntax and semantics and be fluent in the use of Python control flow statements.
2. Express proficiency in the handling of strings and functions.
3. Determine the methods to create and manipulate Python programs by utilizing the data structures like lists, dictionaries, tuples, and sets.
4. Identify the commonly used operations involving file systems and regular expressions. Articulate the Object-Oriented Programming concepts such as encapsulation, inheritance, and polymorphism as used in Python.

UNIT I

INTRODUCTION TO PYTHON:

Python installation, Python syntax, Scripts, Native DataTypes, Booleans, Numbers, Lists, Tuple, Sets, Dictionaries, Comprehensions, List Comprehensions, Dictionary Comprehensions, Set Comprehensions
(12 Lectures)

UNIT II

STRINGS AND MODULES String:

operation, Formatting, Bytes, Encoding, Regular Expressions, Verbose, module declaration, Importing modules, Objects, and Indenting as Requirement, Exceptions, Unbound Variables, Lambda Functions and map
(12 Lectures)

UNIT III

CLASSES Creating classes,

instance methods, Instance Variables, Closures, Generators, Iterators, Assert, Generator Expressions
(12 Lectures)

UNIT IV

TESTING AND FILES Reading and Writing Text Files,

Binary Files, Stream Objects, Standard Input, Output, and Error modes, with statement
(12 Lectures)

UNIT V

GUI IN PYTHON Components and events, root component, entry widgets, text widgets, check buttons, Serializing Objects, Pickle Files, Debugging, Introduction to Django framework

(12 Lectures)

Text and Reference Books:

- 1) Dive into Python, Mark Pilgrim, Press.
- 2) Python: The Complete Reference, Martin C. Brown.
- 3) Fluent Python: Clear, Concise, and Effective Programming (1st Edition), Luciana Ramalho, O'Reilly.
- 4) Learning Python: Powerful Object-Oriented Programming, Mark Lutz, O'Reilly, Shroff Publishers And Distributors.

Suggested E-resources:

1. Introduction to Python (w3schools.com)
2. string — Common string operations — Python 3.11.5 documentation
3. Getting Started With Testing in Python – Real Python

BCA VSEMESTER

Course Code : BCA7101P
Title of the Course : Python Programming Lab
Level of the Course : NHEQF Level 5.5
Credit of the Course : 2
Type of the Course : Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course : Practical 60.
Prerequisite: Experience with a high-level language (C/C++, Java) is suggested. Prior knowledge of a scripting language (Perl, UNIX/Linux shells), and Object-Oriented concepts are helpful but not mandatory.

Objective of the course:

1. To learn how to design and program Python applications.
2. To learn how to use lists, tuples, and dictionaries in Python programs.
3. To learn how to identify Python object types.
4. To define the structure and components of a Python program.
5. To learn how to write loops and decision statements in Python.
6. To learn how to write functions and pass arguments in Python.
7. To learn how to build and package Python modules for reusability.
8. To learn how to read and write files in Python.
9. To learn how to design object-oriented programs with Python classes.
10. To learn how to use exception handling in Python applications for error handling.

Learning Outcomes:

1. Interpret the fundamental Python syntax and semantics and be fluent in the use of Python control flow statements.
2. Express proficiency in the handling of strings and functions.
3. Determine the methods to create and manipulate Python programs by utilizing data structures like lists, dictionaries, tuples, and sets.
4. Identify the commonly used operations involving file systems and regular expressions. Articulate the Object-Oriented Programming concepts such as encapsulation, inheritance, and polymorphism as used in Python.

List of Experiments:

1. Write a Python program/script to create a list and perform the following methods:
1) insert() 2) remove() 3) append() 4) len() (5) pop() 6) clear()
2. Write a Python program/script to create a dictionary and apply the following methods: 1) Print the dictionary items 2) access items 3) use get() 4) change values 5) use len()
3. Write a Python program/script to create a tuple and perform the following methods: 1) Add items 2) len() 3) check for item in tuple 4) Access items
4. Write a Python program to add two numbers, to print a number is positive/negative using if-else, to find the largest number among three numbers.
5. Write a Python Program to read a number and display corresponding day using if-elif-else?
6. Write a menu-driven program to create with the following options: 1. to perform addition 2. to perform subtraction 3. to perform multiplication 4. to perform division
7. Python Program to Implement Conditionals and Loops
8. Python program to use functions and modules
9. Python program to read and write data from & to files in Python
10. Python program based on Classes and Objects
11. Python program based on inheritance
12. Python program based on lambda Function

Text and Reference Books:

1. Dive into Python, Mark Pilgrim, Press.
2. Python: The Complete Reference, Martin C. Brown.

3. *Fluent Python: Clear, Concise, and Effective Programming (1st Edition)*, Luciana Ramalho, O'Reilly.
4. *Learning Python: Powerful Object-Oriented Programming*, Mark Lutz, O'Reilly, Shroff Publishers And Distributors.

Suggested E-resources:

1. Introduction to Python (w3schools.com)
2. string — Common string operations — Python 3.11.5 documentation
3. Getting Started With Testing in Python – Real Python

Scheme of EoSE: -

- 1) Exercise(s): 45
- 2) Viva Voce: 20
- 3) Evaluation of record book: 15

BCA VSEMESTER

Course Code	: BCA7102T
Title of the course	: Cloud Computing
Level of the Course	: NHEQF Level 5.5
Credit of the Course	: 4
Type of the Course	: Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course	: Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.
Prerequisites	: NONE

Objectives of the Course :

1. To provide students with the fundamentals and essentials of Cloud Computing.
2. To provide students a sound foundation of the Cloud Computing so that they are able to start using and adopting Cloud Computing services and tools in their real-life scenarios.
3. To enable students to explore some important cloud computing-driven commercial systems and applications.
4. To expose the students to frontier areas of Cloud Computing and information systems while providing sufficient foundation to enable further study and research.

Learning Outcomes:

1. Explain the core concepts of the cloud computing paradigm: how and why this paradigm shift came about.
2. Characteristics, advantages, and challenges brought about by the various models and services in cloud computing.
3. Apply the fundamental concepts in data centers to understand the trade-offs in power, efficiency, and cost.
4. Identify resource management fundamentals, i.e., resource abstraction, sharing and sandboxing, and outline their role in managing infrastructure in cloud computing.
5. Analyze various cloud programming models and apply them to solve problems on the cloud.

UNIT I

Introduction Cloud Computing: Definition, Types of Clouds, Layer & Services models, Deployment models, Cloud Computing Architecture & infrastructure: Cloud Reference Model, Virtualization: Definition, Types of virtualizations (Compute, Network, Storage), Types of Hypervisor

(12 Lectures)

UNIT II

Cloud Platforms in Industry: Major vendors and their offerings, Introduction to Microsoft Azure, Amazon Web Services (EC2, S3, etc.), Google App Engine. Integration of Private and Public Clouds Cloud applications: Protein structure prediction, Data Analysis, Satellite Image Processing, CRM and ERP, Social networking. Cloud Application - Scientific Application, Business Application.

(12 Lectures)

UNIT III

Advance Topic in Cloud Computing: Cloud Security, Risks and Approaches of Migration into Cloud. Federated Cloud/ Intercloud, Third Party Cloud Services, Business Continuity and Disaster Recovery, Service Level Agreement (SLA), Dynamic resource provisioning and management, Server consolidation and placement policies, Energy efficiency in data centers, Elastic Load Balancing and AutoScaling.

(12 Lectures)

UNIT IV

Storage Network Design: Architecture of storage, analysis and planning. Storage network design considerations; NAS and FC SANs, hybrid storage networking technologies (iSCSI, FCIP, FCoE), design for storage virtualization in cloud computing, host system design considerations. Techniques for Big data processing (Google GFS, BigTable, and Map-Reduce Hadoop Distributed File System (HDFS), HIVE).

(12 Lectures)

UNIT V

Consensus in Cloud Computing: Issues in consensus, Consensus in synchronous and asynchronous system, Byzantine Agreement: Agreement, Faults, Tolerance, Measuring Reliability and Performance, SLIs, SLOs, SLAs, TLAs, Byzantine failure, Byzantine Generals Problem, Failures & Recovery Approaches in Distributed Systems, Checkpointing.

(12 Lectures)

Text and Reference Books:

1. Distributed and Cloud Computing, Kai Hawang, Geoffrey C. Fox, Jack J. Dongarra, Elsevier
2. Cloud Computing Bible, Barrie Sosinsky, Wiley-India, 2010
3. Cloud Computing, Kumar Saurabh, Wiley Pub
4. Cloud Security, Krutz, Vines, Wiley Pub
5. Cloud Computing - A Practical Approach, Velte, TMH

Suggested E-resources:

1. https://onlinecourses.nptel.ac.in/noc21_cs14/preview
2. https://www.youtube.com/watch?v=HhStJ7FDBpc&ab_channel=5MinutesEngineering
3. https://www.cs.cmu.edu/~msakr/15619-s16/15319_15619_s16_Syllabus.pdf

BCA VSEMESTER

Course Code	: BCA7102P
Title of the course	: Cloud Computing Lab
Level of the Course	: NHEQF Level 5.5
Credit of the Course	: 2
Type of the Course	: Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course	: Practical 60.
Prerequisites	: NONE

Objectives of the course

1. To learn and use version control systems
2. To develop web applications in the cloud
3. To learn and work with virtual machines
4. To learn the design and development process involved in creating a cloud-based application
5. To learn to implement and use parallel programming using Hadoop

Learning Outcomes: At the end of the course, learners will be able to:

1. Configure various virtualization tools such as VirtualBox, VMware Workstation.
2. Design and deploy a web application in a PaaS environment.
3. Learn how to simulate a cloud environment to implement new schedulers.
4. Install and use a generic cloud environment that can be used as a private cloud.
5. Install and use Hadoop

List of Experiments

1. Install Virtualbox/VMware Workstation with different flavours of linux or windows OS on top of windows7 or 8.
2. Install a C compiler in the virtual machine created using virtual box and execute Simple Programs
3. Install Google App Engine. Create hello world app and other simple web applications using python/java.
4. Use GAE launcher to launch the web applications.
5. Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.
6. Find a procedure to transfer the files from one virtual machine to another virtual machine.
7. Find a procedure to launch virtual machine using trystack (Online Openstack Demo Version)
8. Install Hadoop single node cluster and run simple applications like word count

Text and Reference Books:

1. Distributed and Cloud Computing, Kai Hawang, Geoffrey C. Fox, Jack J. Dongarra, Elsevier
2. Cloud Computing Bible, Barrie Sosinsky, Wiley-India, 2010
3. Cloud Computing, Kumar Saurabh, Wiley Pub
4. Cloud Security, Krutz, Vines, Wiley Pub
5. Cloud Computing - A Practical Approach, Velte, TMH

Suggested E-resources:

1. https://onlinecourses.nptel.ac.in/noc21_cs14/preview
2. https://www.youtube.com/watch?v=HhStJ7FDBpc&ab_channel=5MinutesEngineering
3. https://www.cs.cmu.edu/~msakr/15619-s16/15319_15619_s16_Syllabus.pdf

Scheme of EoSE: - 1) Exercise(s): 45 2) Viva Voce: 20 3) Evaluation of record book: 15

BCA V SEMESTER

Course Code	: BCA7103T
Title of the course	: Discrete Mathematics Structural
Level of the Course	: NHEQF Level 5.5
Credit of the Course	: 4
Type of the Course	: Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course	: Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.
Prerequisites	: None

Objectives of the Course:

1. In particular, it teaches students how to think logically and mathematically through five important themes: Mathematical reasoning, Combinational Analysis discrete structures, Algorithmic thinking, Applications and modeling.

Learning Outcomes:

1. Propositional Logic prevents from making inconsistent inferences and incautious decisions.
2. Vectors are primarily used within the programming context of most programming languages and serve as data structure containers.
3. Numerical Analysis analyze implements algorithms for obtaining numerical solutions to problems involving continuous variables.

UNIT-I

Logic and Propositional Calculus: Propositions and compound propositions, Basic logical Operations, Propositions and truth tables, Tautologies and Contradictions, Logical Equivalence, Algebra of Propositions, Conditional and biconditional Statements, Arguments, Logical implications, Propositional functions, Quantifiers (12 Lectures)

UNIT-II

Vectors, Matrices, Matrix Addition and Scalar Multiplication, Matrix Multiplication, transpose, Transpose, Square Matrices, Invertible (Nonsingular) Matrices, Inverses, Determinants (12 Lectures)

UNIT-III

Algebraic Systems: Operations, Semigroups, Groups, Subgroups, Normal Subgroups and Homomorphisms, Rings, Integral Domains, and fields, Polynomial over a field. (12 Lectures)

UNIT-IV

Languages, Grammars, Machines: Alphabet, Words, Free Semigroup, Languages, Regular Expressions, Regular languages, Finite state Automata, Grammars, Finite state Machine. (12 Lectures)

UNIT-V

Numerical Analysis: Roots of Transcendental Equations, Bisection Method, Regula falsi Method, Newton Raphson Method (12 Lectures)

Text and Reference Books:

1. The McGraw Hill Company: Discrete Mathematics by Seymour Lipschutz. Higher Engineering Mathematics: B.V. RAMANA

Suggested E-resources:

1. NPTEL

2. Coursera

Scheme of EoSE :

(i)Exercise(s):45

(ii)Viva Voce: 20

(iii) Evaluation of record book: 15

BCA V SEMESTER

Code of the course	: BCA7103P
Title of the course	: Discrete Mathematics Structural Lab
Level of the Course	: NHEQF Level 5.5
Credit of the Course	: 2
Type of the Course	: Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course	: Practical 60.
Prerequisites	: Problem Solving using Core equivalent

Objectives of the Course : The course allows us in storing data while maintaining the data's correctness and efficiency in a computer program.

Learning Outcomes : Numerical Analysis - analyze implements algorithms for obtaining numerical solutions to problems involving continuous variables.

List of Experiments :

1. Write a C program to find a root of the Equation $x^3 - 3x - 5 = 0$ correct to three decimal places by regular falsi method.
2. Write a C program to find a root of the Equation $\cos x - xe^x = 0$ correct to three decimal places by regular falsi method.
3. Write a C program to find a root of the Equation $x^2 - \log_{10}x - 12 = 0$ correct to three decimal places by regular falsi method.
4. Write a C program to find correct to four decimal places the root between 0 and 1 of the equation $x^3 - 6x + 4 = 0$ by Newton Raphson Method.
5. Write a C program to find correct to four decimal places the root between 0 and 1 of the equation $x^3 - 5x + 3 = 0$ by Newton Raphson Method.
6. Write a C program to find correct to four decimal places the root between 0 and 1 of the equation $3x - \cos x - 1 = 0$ by Newton Raphson Method.
7. Write a C program to find the roots of the Equation $x^3 - 4x + 1 = 0$ by Bisection Method.
8. Write a C program to find the roots of the equation $x^4 - x - 10 = 0$ by Bisection Method.
9. Write a C program to find the roots of the equation $x^3 - x - 1 = 0$ by Bisection Method.
10. Write a C program to find the roots of the equation $x^4 + 2x^3 - x - 1 = 0$ by Bisection Method.

Text and Reference Books:

1. Mathematical Methods by G.S. Rao
2. Higher Engineering Mathematics: BVRAMANA

Suggested E-resources:

1. NPTEL
2. Coursera

Scheme of EoSE :

(i)Exercise(s):45 (ii)Viva Voce: 20 (iii) Evaluation of record book: 15

BCA V SEMESTER

Code of the course	: BCA7104T
Title of the Course	: Internet of Things
Level of the Course	: NHEQF Level 5.5
Credit of the Course	: 4
Type of the Course	: Discipline Specific Elective (DSE) Course in Computer Science

Delivery Type of the Course : Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.

Prerequisites :

1. Must know how systems work.
2. To know about statistics that have to do with Machine Learning.
3. Simultaneous use of artificial intelligence.
4. Need to know something about sensors.
5. Basic Knowledge of Networking protocols.

Objectives of the course : To obtain and analyze data from things (devices) that were previously disconnected from most data processing tools.

Learning Outcomes :

1. Explain the definition and usage of the term “Internet of Things” in different contexts.
2. Understand the key components that make up an IoT system.
3. Differentiate between the levels of the IoT stack.
4. Familiarity with the key technologies and protocols employed at each layer of the IoT stack.
5. Able to build IoT applications.
6. Learn to integrate modern technologies such as sensors, communication, and computational processing.

UNIT I

Introduction to IoT: Internet of Things - Physical Design - Logical Design - IoT Enabling Technologies - IoT Levels & Deployment Templates - Domain Specific IoTs - IoT and M2M - IoT System Management with NETCONF-YANG - IoT Platforms Design Methodology

(12 Lectures)

UNIT II

IoT Architecture - M2M high-level ETSI architecture - IETF architecture for IoT - OGC architecture - IoT reference model - Domain model - information model - functional model - communication model - IoT reference architecture

(12 Lectures)

UNIT III

IoT Protocols - Protocol Standardization for IoT, Efforts, M2M and WSN Protocols, SCADA and RFID Protocols, Unified Data Standards, Protocols, IEEE 802.15.4, BACNet Protocol, Modbus–Zigbee Architecture, Network layer, 6LowPAN-CoAP-Security

(12 Lectures)

UNIT IV

Building IoT With Raspberry Pi & Arduino - Building IoT with RASPBERRY PI - IoT Systems - Logical Design using Python, IoT Physical Devices & Endpoints - IoT Device - Building blocks - Raspberry Pi - Board - Linux on Raspberry Pi - Raspberry Pi Interfaces - Programming Raspberry Pi with Python - Other IoT Platforms – Arduino

(12 Lectures)

UNIT V

Case Studies and Real-World Applications - Real-world design constraints - Applications - Asset management, Industrial automation, smart grid, Commercial building automation, Smart cities - participatory sensing - Data Analytics for IoT, Software & Management Tools for IoT Cloud Storage Models & Communication APIs - Cloud for IoT - Amazon Web Services for IoT

(12 Lectures)

Text and Reference Books:

1. Internet of Things, A hands-on approach by Arshdeep Bahga, Vijay Madisetti, Universities Press.
2. Architecting the Internet of Things by Dieter Uckelmann, Mark Harrison, Michahelles, Florian (Eds), Springer.
3. The Internet of Things in the Cloud: A Middleware Perspective by Honbo Zhou, CRC Press.
4. The Internet of Things, Key applications and Protocols by Olivier Hersent, David Boswarthick, Omar Elloumi, Wiley.

Suggested E-resources:

1. NPTEL
2. Coursera

BCA V SEMESTER

Code of the course : BCA7104P
Title of the course : Internet of Things Lab
Level of the Course : NHEQF Level 5.5
Credit of the Course : 2
Type of the Course : Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course : Practical 60. The student will perform the experiments and submit the record of observations after getting the results.

Prerequisites :

It includes a strong foundation in mathematics and statistics, programming proficiency (languages like Python), and a solid understanding of machine learning concepts. Also, Data manipulation and analysis, linear algebra and calculus, is beneficial.

Objectives of the Course :

The primary goal of artificial intelligence is to develop a technology that will capacitate computer systems to perform independently of human intervention and intelligently. It improves problem-solving skills and includes knowledge representation. It encourages human-AI collaboration.

Learning Outcomes:

1. To apply Control Structures and Operators to write basic Python programming.
2. To analyze object-oriented concepts in Python.
3. To evaluate the AI models pre-processed through various feature engineering algorithms by Python Programming.
4. To create the code for the recommender system using Natural Language processing.
5. To create various reinforcement algorithms to solve real-time complex problems.

List of Experiments:

1. Write a python program to print the multiplication table for the given number?
2. Write a python program to check whether the given number is prime or not?
3. Write a python program to find the factorial of the given number?
4. Write a python program to implement a simple Chatbox?
5. Write a python program to implement List operations (Nested List, Length, Concatenation, Membership, Iteration, Indexing, and Slicing)?
6. Write a python program to implement List methods (Add, Append, Extend, and Delete).
7. Write a python program to Illustrate Different Set Operations?
8. Write a python program to generate a Calendar for the given month and year?
9. Write a python program to implement a Simple Calculator program?
10. Write a python program to Add Two Matrices.
11. Write a python program to Transpose a Matrix.
12. Write a python program to remove punctuations from the given string?
13. Write a python program to sort the sentence in alphabetical order?
14. Write a python program to compute Mean, Median, Mode, Variance, and Standard Deviation using Datasets.
15. Write a python program to implement Breadth-First Search Traversal?
16. Write a python program to implement Depth-First Search Traversal?
17. Write a python program to implement the Water Jug Problem?
18. Write a Program to Implement the Travelling Salesman Problem using Python.
19. Write a Program to implement 8-Queens Problem using Python.
20. Write a program to implement Tic-Tac-Toe game using python.
21. Write a Program to Implement 8-Puzzle problem using Python.
22. Write a Program to Implement Tower of Hanoi using Python.

Text and Reference Books:

1. “Artificial Intelligence: A Modern Approach”, Stuart J. Russell & Peter Norvig, Pearson Education.
2. “Introduction to AI & Expert System”, Dan W. Patterson, PHI.
3. “Artificial Intelligence – Structure and Strategies of Complex Problem Solving”, George F. Luger, Pearson Education.
4. “Artificial Intelligence”, Elaine Rich, Kevin Knight, Mc-Graw Hill.

Suggested E-resources:

1. NPTEL and Coursera

Scheme of EoSE :

(i)Exercise(s):45

(ii)Viva Voce: 20

(iii) Evaluation of record book: 15

BCA V SEMESTER

Code of the course	: BCA7105T
Title of the course	: Desktop Publishing
Level of the Course	: NHEQF Level 5.5
Credit of the Course	: 4
Type of the Course	: Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course	: Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.
Prerequisites	: None

Objectives of the Course :

The course allows us in storing data while maintaining the data's correctness and efficiency in a computer program.

Learning Outcomes:

1. After successful completion of the module, the students shall be able to: Know and use every MS Paint available option for professional or personal use.
2. Use Photoshop as a premier graphic design and image editing tool . Productively utilize the Photoshop environment . Effectively utilize the multiple methods of manipulating, retouching, refurbishing art work, photographs . Introduced to the advanced theories of mask, alpha channel and clipping path .Gain entry level position in Graphic design and animation . Understand multiple image types and the best application of each for graphic design.
3. PageMaker- Create Documents and Templates, add text into documents using various methods, and apply different formatting styles to characters and paragraphs. Import graphics, create objects using various tools, add effects to objects .Create a book and export it into PDF Multipage Layout Design
4. CorelDraw: After successful completion of this module, students should be able to: Conceptualize and create Logos, various types of print designs. Design Pamphlets, Posters, Invitation cards, Greeting cards, Wrappers, Advertisements, Banners and Package. Create multifaceted drawings with several layers and details Add a great degree of dimensional effect and richness to drawings.

Unit I

MS Paint: Introduction About the limits of MS Paint ,Presentation and setup of user interface and help ,Open and save an image , Knowledge of available file types (JPG, TIFF, ICO, PNG, GIF...) , Set opened image as desktop wallpaper , Display options (zoom, miniature, grid, etc.) , Define or resize the size of an image (nonfunctional transparency) , Drawing tools overview , Colors selection with right click/left click in the palette ,Copy/Paste from selection with or without transparency , Insert an external image in a composition , Colors number selection and color inversion

(12 Lectures)

UNIT II

Photoshop: Getting Acquainted with Photoshop , Basic Image Manipulation , Color Basics Painting Tools , Brush Settings , Making Selections ,Filling and stroking , Layers , Advanced Layers , Text , Drawing , Using Channels and Masking , Manipulating images , Getting to know the work area , Basic Photo Corrections , Retouching and Repairing ,Working with selections , Layer Basics , Masks and channels , Correcting and enhancing digital photographs , Vector drawing techniques , Producing and printing

(12 Lectures)

UNIT III

PageMaker: Getting Started with PageMaker, PageMaker Interface, Creating a New Document, Managing Document Layer, Creating & Editing Text, Working with Edit Story, Managing Text as an Object , Working with Text and Graphics, Using Graphics, Applying Colors to Graphics, Framing Graphics,

Cropping and masking Graphics, Working with Layers, Working with Master pages, Working with Plugins, Using Text Wrap , Color Separation Capabilities, Printing

(12 Lectures)

UNIT IV

Coral Draw: Getting started with Corel Draw, Introduction to Corel Draw, Features of Corel Draw, Corel Draw Interface, Tool Box, Moving from Adobe Illustrator to Corel Draw, Common Tasks , Drawing and Coloring, Selecting Objects, Creating Basic Shapes, Reshaping Objects, Organizing objects, Applying color fills and Outlines , Mastering with Text, Text Tool Artistic and paragraph text, Formatting Text , Applying Effects, Power of Blends Distortion, Contour Effects, Working with Bitmap Commands, Working with Bitmaps, Editing Bitmaps, Applying effects on Bitmaps, Printing

(12 Lectures)

Text and References Books:

1. Thinking with Type
2. Desktop Publishing

Suggested E-resources

1. https://www.academia.edu/66235623/Desktop_publishing_a_critical_bibliography

BCAVSEMESTER

Code of the course	: BCA7105P
Title of the course	: Desktop Publishing Lab
Level of the Course	: NHEQF Level 5.5
Credit of the Course	: 2
Type of the Course	: Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course	: Practical 60.
Prerequisites	: None

Objectives of the Course :

The course allows us in storing data while maintaining the data's correctness and efficiency in a computer program.

Learning Outcomes :

1. This course introduces students to the principles of design applicable to publications created using desktop publishing software and computer technology.
2. The students will discover how to use the essential building blocks of design type, art and line in new and creative ways, learn clever ways to locate and use resources such as graphics and scanned art, learn to think about audience and medium and how those affect the way you craft your message and also be learning to use new technical tools.

List of Programs based on following software's

1. MS Paint
2. Photoshop
3. PageMaker
4. CorelDraw

Text and References Book:

1. Adobe Photoshop CC Classroom in a Book – Adobe creative team, Adobe press
2. Adobe PageMaker 7.0 Classroom in a Book – Adobe creative team, Adobe press
3. CorelDraw X8: The official guide Gary David Bouton

Suggested E-resources.

1. https://www.academia.edu/66235623/Desktop_publishing_a_critical_bibliography

Scheme of EoSE :

(i)Exercise(s):45

(ii)Viva Voce: 20

(iii) Evaluation of record book: 15

BCA V SEMESTER

Code of the course : BCA7106T
Title of the course : Mobile Application Development
Level of the Course : NHEQF Level 5.5
Credit of the Course : 4
Type of the Course : Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course : Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.

Prerequisites :
Prior knowledge of basic computer programming and logic Building is essential for learning Mobile Application Development.

Objectives of the Course:

The course aims to understand the concepts of developing Android application with User interface, networking and animation.

Learning Outcomes:

1. To explain the basics of mobile application development
2. Develop Android application with User interface, networking and animation.
3. Use simulator tools to test and publish the application.
4. Understand how data can be transferred using XML.
5. Understand setting up of Android environment.

UNIT-I

Mobile Application Development - Mobile Applications and Device Platforms, Alternatives for Building Mobile Apps, Comparing Native vs. Hybrid Applications, The Mobile Application Development Lifecycle, The Mobile Application Front-End, The Mobile Application Back-End, Key Mobile Application Services, What is Android, Android version history, Obtaining the Required Tools, Launching Your First Android Application, Exploring the IDE, Debugging Your Application, Publishing Your Application
(12 Lectures)

UNIT-II

Understanding Activities - Linking Activities Using Intents, Fragments, Displaying Notifications, Understanding the Components of a Screen, Adapting to Display Orientation, Managing Changes to Screen Orientation, Utilizing the Action Bar, Creating the User Interface Programmatically Listening for UI Notifications
(12 Lectures)

UNIT-III

Using Basic Views - Using Picker Views, Using ListView to Display Long Lists, Understanding Specialized Fragments, Using Image Views to Display Pictures, Using Menus with Views, Using Web View, Saving and Loading User Preferences, Persisting Data to Files, Creating and Using Databases.
(12 Lectures)

UNIT-IV

Sharing Data in Android - Creating Your Own Content Providers, Using the Content Provider, SMS Messaging, Sending Email, Displaying Maps, Getting Location Data, Monitoring a Location.
(12 Lectures)

UNIT-V

Consuming Web Services Using HTTP, Consuming JSON Services, Creating Your Own Services, Binding Activities to Services, Understanding Threading.
(12 Lectures)

Text and Reference Books

1. Jerome DiMarzio, “Beginning Android Programming with Android Studio”, 4th Edition.
2. Dawn Griffiths, David Griffiths, “Head First Android Development: A Brain-Friendly Guide”, 2017.
3. Neil Smyth, “Android Studio 3.0 Development Essentials: Android”, 8th Edition.
4. Pradeep Kothari, “Android Application Development (With Kitkat Support)”, Black Book 2014.

Suggested E-resources.

1. https://en.wikipedia.org/wiki/Mobile_app_development
2. <https://flutter.dev/>
3. https://en.wikipedia.org/wiki/Android_10/

BCA V SEMESTER

Code of the course : BCA7106P
Title of the course : Mobile Application Development LAB
Level of the Course : NHEQF Level 5.5
Credit of the Course : 2
Type of the Course : Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course : Practical 60.

Prerequisites:

Prior knowledge of basic computer programming and logic Building is essential for learning Mobile Application Development.

Objectives of the Course:

1. Know the components and structure of mobile application development frameworks for Android and windows OS based mobiles.
2. Understand how to work with various mobile application development Frameworks.
3. Learn the basic and important design concepts and issues of development of mobile applications.
4. Understand the capabilities and limitations of mobile devices.

Learning Outcomes:

1. Develop mobile applications using GUI and Layouts.
2. Develop mobile applications using Event Listener.
3. Develop mobile applications using RSS Feed, Internal/External Storage, SMS, Multi-threading
4. Analyze and discover own mobile app for simple needs.

List of sample programs

1. Develop an application that uses GUI components, Font and Colours.
2. Develop a native calculator application.
3. Develop an application that finds greatest among three numbers using GUI Components.
4. Develop an application to display your personal details using GUI Components.
5. Develop an application that uses the radio button.
6. Develop an application that uses the image button.
7. Develop an application that uses Alert Dialog Box.
8. Develop an application that makes use of Notification Manager.
9. Implement an application that creates an alert upon receiving a message.
10. Write an application that draws basic graphical primitives on the screen.
11. To understand Activity, Intent, Create sample application with login module.(Check username and password).
12. Develop an application to set an image as wallpaper. On click of a button, the wallpaper image should start to change randomly every 30 seconds

Suggested Books and References:

1. David Mark, Jack Nutting, Jeff LaMarche and Frederic Olsson, "Beginning iOS 6 Development: Exploring the iOS SDK", Apress, 2013.
2. James Dovey and Ash Furrow, "Beginning Objective C", Apress, 2012

Suggested E-resources.

- (1) https://en.wikipedia.org/wiki/Mobile_app_development
- (2) <https://flutter.dev/>
- (3) https://en.wikipedia.org/wiki/Android_10/

Scheme of EoSE :

(i)Exercise(s):45 (ii)Viva Voce: 20 (iii) Evaluation of record book: 15

BCA V SEMESTER

Code of the course	: BCA7107T
Title of the course	: Advance Web Development
Level of the Course	: NHEQF Level 5.5
Credit of the Course	: 4
Type of the Course	: Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course	: Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.

Objectives of the Course

1. The increasing practice of Web-based applications, this course focuses on Advanced Web Technologies with the concept of Laravel Framework
2. This subject will attempt to give basic understanding of HTML, CSS, Javascript, URL Methods, MVC Framework, PHP, Laravel

Learning Outcomes:

1. Develop a static, interactive and well-formed webpage using JavaScript, CSS3 and HTML5.
2. Use PHP7 to improve accessibility of a web document.
3. Gain necessary skills for designing and developing web applications.

Unit-I

Introduction: Basics of WWW, HTTP protocol methods and headers, HTTP Request and Response, Architecture of web browser, Web server installation and configuration, Web security, CORS, Understanding SEO HTML: HTML page structure, formatting tags in HTML, tables, links, images, meta tags, frames, html form tags, media, APIs, HTML5 tags in relation to validations and SEO

(12 Lecture)

Unit-II

CSS: Need for CSS, Basic syntax and structure, Backgrounds, Colors and properties, Manipulating texts, Fonts, borders and boxes, Margins, Padding Lists, CSS2, CSS3, Animations, Tool-Tips, Style images, Variables, Media Queries, Wildcard Selectors (, ^ and \$) in CSS, Media Query, CSS variables

(12 Lecture)

Unit-III

JAVASCRIPT: Javascript Syntax, Types of Javascript, variables, arrays, functions, conditions, loops, Popup boxes, Javascript objects and DOM, Javascript inbuilt functions, Javascript validations, Regular expressions, Event handling with Javascript, Callbacks in Javascript, Function as arguments in Javascript, Object concepts in Javascript, JSON

(12 Lecture)

Unit-IV

Php Programming: Introduction to Server-side programming ,PHP variables, decision and looping with examples, PHP and HTML, Arrays, Functions, Browser control and detection, String, Form processing, File uploads, Dates and timezone, Working with Regular Expressions, Exception Handling, Working with JSON data, Object Oriented Programming with PHP

(12 Lecture)

Unit-V

PHP MVC Framework Laravel Introduction to Laravel and MVC, Environment Setup, Routes, Namespaces, Controllers, Blade template, Views, Request Response, Redirections, Forms, Session, Cookies, Migration, Error handling, Database connectivity and CRUD Operations.

(12 Lecture)

Text and Reference Books:

1. HTML and CSS: Design and Build Websites, 1st Edition - Jon Duckett
2. JavaScript and jQuery: Interactive Frontend Web Development, 1st Edition

3. Eloquent JavaScript, 3rd Edition: A Modern Introduction to Programming

Suggested E-resources:

1. How to use PHP And MySQL in Web Development? (step-by-step guide) (knowledgehut.com)
2. Learn JavaScript for Web Development | Udemy
3. CSS Tutorial ([w3schools.com](https://www.w3schools.com))

BCA V SEMESTER

Code of the course : BCA7107T
Title of the course : Advance Web Development Lab
Level of the Course : NHEQF Level 5.5
Credit of the Course : 2
Type of the Course : Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course : Practical 60.

Objectives of the Course :

Learn programming languages and technologies: Web development courses typically focus on teaching programming languages such as HTML, CSS, and JavaScript, Php as well as frameworks and libraries like Laravel Framework. The objective is to provide a solid foundation in these languages and technologies.

Learning Outcomes :

(1) Develop a static, interactive and well-formed webpage using JavaScript, CSS3 and HTML5.
(2) Use PHP7 to improve accessibility of a web document. (3) Gain necessary skills for designing and developing web applications.

Practicals

HTML And CSS HTML: HTML page structure, formatting tags in HTML, tables, links, images, meta tags, frames, html form tags, media, APIs, HTML5 tags in relation to validations Need for CSS, Basic syntax and structure, Backgrounds, Colors and properties, Manipulating texts, Fonts, borders and boxes, Margins, Padding Lists, CSS2, CSS3, Animations, Tool-Tips, Style images, Variables, Media Queries, Wildcard Selectors (, ^ and \$) in CSS, Media Query, CSS variables

JavaScript and Jquery JavaScript Introduction, Variable declaration (Let, var and Const Keyword), Operators, Control Statements, Understanding arrays, Function Declaration, Built-In Functions, Standard Date and Time Functions

HTML Document object Model, Working with HTML form and its elements, HTML Document object Model, Working with HTML form and its elements, Other Document Object Model, Events In JavaScript,

jQuery Introduction, jQuery Function, \$.post(), \$.get() Php Programming: Introduction to Server-side programming ,PHP variables, decision and looping with examples, PHP and HTML, Arrays, Functions, Browser control and detection, String, Form processing, File uploads, Dates and time zone, Working with Regular Expressions, Exception Handling, Working with JSON data, Object Oriented Programming with PHP

PHP MVC Framework Laravel Introduction to Laravel and MVC, Environment Setup, Routes, Namespaces, Controllers, Blade template, Views, Request Response, Redirections, Forms, Session, Cookies, Migration, Error handling, Database connectivity and CRUD Operations.

Text and Reference Books:

1. HTML and CSS: Design and Build Websites, 1st Edition - Jon Duckett
2. JavaScript and jQuery: Interactive Frontend Web Development, 1st Edition
3. Eloquent JavaScript, 3rd Edition: A Modern Introduction to Programming

Suggested E-resources:

1. How to use PHP And MySQL in Web Development? (step-by-step guide) (knowledgehut.com)
2. Learn JavaScript for Web Development | Udemy
3. CSS Tutorial (w3schools.com)

Scheme of EoSE :

(i)Exercise(s):45 (ii)Viva Voce: 20 (iii) Evaluation of record book: 15

DSE Group – C1

BCA SEMESTER

Code of the course	: BCA7108T
Title of the course	: Artificial Intelligence
Level of the Course	: NHEQF Level 5.5
Credit of the Course	: 4
Type of the Course	: Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course	: Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.
Prerequisites	: None

Objectives of the Course :

1. To develop systems that can analyze large datasets, identify patterns, and make data-driven decisions.
2. This ability to solve problems and make decisions efficiently is invaluable across various industries, from healthcare and finance to transportation and manufacturing.

Learning Outcomes:

1. Analyze the implications of applying AI systems to organizations and the future of work.
2. Explain how to develop AI systems to meet business, organizational, and technology requirements.
3. Implement AI frameworks and platforms to improve business, organizational, and technology outcomes.

Unit – I

Introduction: Introduction to Artificial Intelligence, Background and Applications, AI techniques, Tic-Tac-Toe problem, Problem Characteristics.

(12 Lecture)

Unit – II

Problem Solving and Searching Techniques: Problem Characteristics, Production Systems, Water Jug Problem, Control Strategies, Breadth First Search, Depth First Search, Hill climbing and its Variations, Heuristic Search Techniques: Best First Search. 8-Puzzle Problem.

(12 Lecture)

Unit – III

Knowledge Representation: Definition of Knowledge, Knowledge-Based Systems, Representation of Knowledge. Introduction to First Order Predicate Logic, Conversion to clausal form, Unification, Resolution Principle.

(12 Lecture)

Unit – IV

Expert Systems: Introduction to Expert Systems, Characteristic Features of Expert Systems, Applications of Expert Systems, Components and Working of Expert Systems

(12 Lecture)

Unit – V

Introduction to Machine Learning Techniques: Fuzzy Logic, Fuzzy Set, Membership Function, Union, intersection and complement of a fuzzy set, Introduction to Artificial Neural Network, Introduction to Support Vector Machine.

(12 Lecture)

Text and Reference Book:

1. DAN.W.Patterson, Introduction to A.I and Expert Systems – PHI, 2007.
2. Russell & Norvig, Artificial Intelligence - A Modern Approach, LPE, Pearson Prentice Hall, 2nd edition, 2005.
3. Rich & Knight, Artificial Intelligence – Tata McGraw Hill, 2nd edition, 1991.
4. W.F.Clocksın and Mellish, Programming in PROLOG, Narosa Publishing House, 3rd edition, 2001.

Suggested E-resources:

1. <https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-034-artificial-intelligence-fall-2010/lecture-videos/>
2. <https://nptel.ac.in/courses/106/102/106102220/>
3. <https://nptel.ac.in/courses/106/105/106105078/>

BCA SEMESTER

Code of the course : BCA7108P
Title of the course : Artificial Intelligence Lab
Level of the Course : NHEQF Level 5.5
Credit of the Course : 2
Type of the Course : Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course : Practical 60.

Prerequisites :

Prior knowledge of computer programming and networking basics is essential for learning IoT. Familiar with electronics and sensor technologies, and an understanding of data handling and communication protocols, would be beneficial for grasping IoT concepts effectively.

Objectives of the Course :

The course aims to understand the concepts of Internet of Things and familiarize with Arduino Programming. Also, understand IoT programming and Raspberry Pi Device. We can perform SQL queries in Raspberry Pi. It implements IoT application using Raspberry Pi.

Learning Outcomes :

1. We understand the use of IoT devices for the exchange of data.
2. We understand to use the programming skills of Raspberry Pi.
3. We can develop the cloud platform to analyze and upload sensor data.
4. We can design the applications using Arduino and Raspberry Pi.

Syllabus:

1. Sense the Available Networks using Arduino.
2. Measure the Distance using Ultrasonic Sensor and make LED blink using Arduino.
3. Detect the Vibration of an object using Arduino.
4. Connect with the available Wi-Fi using Arduino.
5. Sense a Finger when it is placed on Board using Arduino.
6. Temperature Notification using Arduino.
7. LDR to vary the Light intensity of LED using Arduino.
8. MySQL Database Installation in Raspberry Pi.
9. SQL Queries by fetching Data from Database in Raspberry Pi.
10. Switch Light on and off Based on the user using Raspberry Pi.

Text and References Book:

1. Vijay Madiseti and Arshdeep Bahga, "Internet of Things: (A Hands-on Approach)", Universities Press (INDIA) Private Limited 2014, 1st Edition.
2. Michael Miller, "The Internet of Things: How Smart TVs, Smart Cars, Smart Homes, and Smart Cities Are Changing the World", Pearson Education 2015.
3. Francis da Costa, "Rethinking the Internet of Things: A Scalable Approach to Connecting Everything", Apress Publications 2013, 1st Edition.
4. Cuno Pfister, "Getting Started with the Internet of Things", O'Reilly Media 2011.

Suggested E-resources.

1. <https://github.com/connectIOT/iottoolkit>
2. <https://www.arduino.cc/>
3. <http://www.zettajs.org/>

Scheme of EoSE : (i)Exercise(s):45 (ii)Viva Voce: 20 (iii) Evaluation of record book: 15

BCA V SEMESTER

Code of the course : BCA7109T
Title of the course : Data Analytics Using Excel
Level of the Course : NHEQF Level 5.5
Credit of the Course : 4
Type of the Course : Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course : Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.

Objectives of the Course :
Master the most popular Excel tools and come out with confidence to complete any Excel tasks with efficiency and grace.

Learning Outcomes :
At the end of this course, the student would be able to understand fundamental and intermediate knowledge of Microsoft Excel 2007, 2010, 2013, 2016, 2019 and Microsoft Office 365.

Unit - I

Microsoft Excel Fundamental:

(a) Launching Excel, Microsoft Excel Startup Screen, Introduction to the Excel Interface, Customize the Quick Access Toolbar, Understanding the Structure of Excel workbook, Saving an Excel Document, Opening Existing Excel Workbook, Common Excel Shortcut Keys.

(b) Entering and Editing Text and Formulas, working with basic Excel functions, Modifying an Excel Worksheets, Formatting Data in worksheets, Inserting images and shapes, Creating Basic chart, Printing an Excel Worksheets, working with Excel Templates.

(12 Lectures)

Unit - II

Working with an Excel List:

(a) Understanding Excel List Structure, Sorting a List Using Single Level-multilevel sort, Using Custom-sort in Excel List, Filter an Excel List Using the Auto filter Tool, format a List As a table, using conditional Formatting to find duplicates, Removing Duplicates.

(b) Excel List Function: Excel DSUM(), DVERAGE(), DCOUNT(), SUBTOTAL()

(12 Lectures)

Unit - III

Excel Data Validation, Importing and Exporting Data, Excel Pivot Tables, Working with large set of Excel Data

(12 Lectures)

Unit - IV

(a) Working with Excel's conditional functions: IF(), AND(), COUNTIF(), SUMIF(), IFERROR(),

(b) Working with Excels Lookup Functions: LOOKUP, HLOOKUP, VLOOKUP, INDEX-MATCH, combined LOOKUP Functions.

(c) Working with Excel's Text based function: LEFT(), RIGHT(), MID(), LEN(), SEARCH(), CONCATENATE()

(d) Working with basic Excel Function's: SUM(), MIN(), MAX(), AVERAGE(), COUNT()

(12 Lectures)

Unit - V

(e) Protecting Excel Worksheets and Workbooks: Protecting Specific Cell in Worksheets, Protecting the Structure of a workbook, Adding a Workbook Password.

(f) Excel's "What if": Working with Excel's Goal Seek Tool, Solver Tool, and Creating Scenarios.
Automating Repetitive Tasks in Excel with Macros

(12 Lectures)

Text and Reference Book:

1. Microsoft Excel 2019 Data Analysis and Business Modeling By Tim King, Executive Editor at Solutions Review
2. Data Analysis with Microsoft Excel Paperback by K. Berk , Partrick Carey
3. Excel Data Analysis Modeling and Simulation by Hector Guerrero

Suggested E-resources.

1. How to Add Data Analysis In Excel: The Best Guide [2023 Edition] (simplilearn.com)
2. A Comprehensive guide to Microsoft Excel for Data Analysis (analyticsvidhya.com)
3. Protecting Excel Workbooks and Worksheets with Passwords (agitraining.com)

BCA V SEMESTER

Code of the course	: BCA7109P
Title of the course	: Data Analytics Using Excel Lab
Level of the Course	: NHEQF Level 5.5
Credit of the Course	: 2
Type of the Course	: Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course	: Practical 60.

Objectives of the Course :

At completion of this course, you will have mastered the most popular Excel tools and come out with confidence to complete any Excel tasks with efficiency and grace.

Learning Outcomes: At the end of this course, the student would be able to understand fundamental and intermediate knowledge of Microsoft Excel 2007, 2010, 2013, 2016, 2019 and Microsoft Office 365.

Practical Based on the following Content:

Unit - I

Microsoft Excel Fundamental:

(c) Launching Excel, Microsoft Excel Startup Screen, Introduction to the Excel Interface, Customize the Quick Access Toolbar, Understanding the Structure of Excel workbook, Saving an Excel Document, Opening Existing Excel Workbook, Common Excel Shortcut Keys.

(d) Entering and Editing Text and Formulas, working with basic Excel functions, Modifying an Excel Worksheets, Formatting Data in worksheets, Inserting images and shapes, Creating Basic chart, Printing an Excel Worksheets, working with Excel Templates.

(12 Lectures)

Unit - II

Working with an Excel List:

(c) Understanding Excel List Structure, Sorting a List Using Single Level-multilevel sort, Using Custom-sort in Excel List, Filter an Excel List Using the Auto filter Tool, format a List As a table, using conditional Formatting to find duplicates, Removing Duplicates., Excel List Function: Excel DSUM(), DAVERAGE(), DCOUNT(), SUBTOTAL()

(12 Lectures)

Unit - III

Excel Data Validation, Importing and Exporting Data, Excel Pivot Tables, Working with large set of Excel Data

(12 Lectures)

Unit - IV

(i) Working with Excel's conditional functions: IF(), AND(), COUNTIF(), SUMIF(), IFERROR(), Working with Excels Lookup Functions: LOOKUP, HLOOKUP, VLOOKUP, INDEX-MATCH, combined LOOKUP Functions. Working with Excel's Text based function: LEFT(), RIGHT(), MID(), LEN(), SEARCH(), CONCATENATE(), Working with basic Excel Function's: SUM(), MIN(), MAX(), AVERAGE(), COUNT()

(12 Lectures)

Unit - V

(j) Protecting Excel Worksheets and Workbooks: Protecting Specific Cell in Worksheets, Protecting the Structure of a workbook, Adding a Workbook Password. Excel's "What if": Working with Excel's Goal Seek Tool, Solver Tool, and Creating Scenarios. Automating Repetitive Tasks in Excel with Macros

(12 Lectures)

Text and References Book:

1. Microsoft Excel 2019 Data Analysis and Business Modeling By Tim King, Executive Editor at

Solutions Review

2. Data Analysis with Microsoft Excel Paperback by K. Berk , Partrick Carey
3. Excel Data Analysis Modeling and Simulation by Hector Guerrero

Suggested E-resources.

1. How to Add Data Analysis In Excel: The Best Guide [2023 Edition] (simplilearn.com)
2. A Comprehensive guide to Microsoft Excel for Data Analysis (analyticsvidhya.com)
3. Protecting Excel Workbooks and Worksheets with Passwords (agittraining.com)

Scheme of EoSE :

(i)Exercise(s):45

(ii)Viva Voce: 20

(iii) Evaluation of record book: 15

BCA V SEMESTER

Code of the course : BCA7110T
Title of the course : Dot Net Programming
Level of the Course : NHEQF Level 5.5
Credit of the Course : 4
Type of the Course : Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course : Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.

Prerequisites:

1. Problem Solving using C courses of bachelor degree course in computer science or equivalent.

Objectives of the Course:

1. Create web application and use different server control to create rich user and interactive website and use different techniques of state management

Learning Outcomes:

1. Implement Arithmetic calculator in C Program.
2. Display message using Textbox, Label and Button Control
3. Write a program with the help of Radio Button, Checkbox, and Panel image control.
4. Design a Form.
5. Design a Menubar.
6. Design Simple Richtext to perform cut, copy and paste functionality.
7. Design a web form to allow user to enter following details in his Resume using webserver control. Set Validation using properties. When data is submitted it must be viewed in the panel below form. Fields of Resume are : First Name, Surname, Gender, Address, City, Pincode, Phone, Qualification, Specialization, Subject, Percentage.
8. Design Gridview.
9. Perform Insert, Update and Delete operations using SQL Server

UNIT-I

Introduction to .NET Framework and ASP.NET

State the Components of Framework and describe CLR : Microsoft .NET Framework overview, NET Framework Architecture, .NET Framework Components (CLR, CLS, CTS, MSIL, Namespace, JIT, Metadata) o Explain benefits of ASP over classic ASP, The client- Server architecture o Develop applications using ASP.NET IDE o ASP.NET(WEB FORMS) - Develop Simple web page using built in objects : Types of ASP.NET Files, Page Life Cycle

(12 Lectures)

UNIT-II

ASP.NET Controls and Validations: Use controls available with IDE platform of ASP.NET for given purpose Button, Checkbox, Check box List Radio button, Radio Buttonlist Drop down List, Hyperlink ,Image, Image Button,Label,Link Button, List Box, List Item,Panel,Text Box Validation Controls : Required field validation, Range Validation Regular Expression, Custom Validation, Validation Summary

(12 Lectures)

UNIT-III

Styles, Themes and Master Pages: Creating Style sheet, applying style sheet rules o Themes, Basics of Master Page, How master page and content pages are connected o Nesting Master page

(12 Lectures)

UNIT-IV

ASP.NET State Management: State Management : Client Side state management Technique and Server Side Management technique o View State, The Query String, Cookies,Session State, Application State o The Global.asax application files , ASP.NET Configuration files

(12 Lectures)

UNIT-V

Connecting Database Using ADO.NET: ADO.NET Architecture : Data provider, Connection Object , Command Object , DataReader Object , DataAdapter Object o Grid View, Dataset, Data View Data Binding o SQL Data Source : Select, Update , Deleting Records

(12 Lectures)

Text and Reference Book:

1. Web Programming Using ASP.NET (Hitesh Patel, Isha, Hemal) publication Atul prakashan
2. Tata McGraw Hill Education Private Limited (TMH)

Suggested E-resources.

1. NPTEL
2. Coursera

BCA V SEMESTER

Code of the course	: BCA7110P
Title of the course	: Dot Net Programming Lab
Level of the Course	: NHEQF Level 5.5
Credit of the Course	: 2
Type of the Course	: Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course	: Practical 60.

Prerequisites:

1. You should have a basic understanding of Computer Programming terminologies. Must have knowledge.
2. A basic understanding of any of the programming languages will help you in understanding the Math concepts and move fast on the learning track.

Objectives of the Course: Students will have the ability to:

1. This course is designed to provide the knowledge of Dot Net Frameworks along with C#
2. To learn about basic features of ASP.NET and its controls To create an ASP.NET application using standard .NET Controls
3. To learn about connecting data sources using ADO.NET and managing them

Learning Outcomes:

1. Implement Arithmetic calculator in C Program.
2. Display message using Textbox, Label and Button Control
3. Write a program with the help of Radio Button, Checkbox, and Panel image control.
4. Design a Form.
5. Design a Menubar.
6. Design Simple Richtext to perform cut, copy and paste functionality.
7. Design a web form to allow user to enter following details in his Resume using webserver control. Set Validation using properties. When data is submitted it must be viewed in the panel below form. Fields of Resume are : First Name, Surname, Gender, Address, City, Pincode, Phone, Qualification, Specialization, Subject, Percentage.
8. Design Gridview.
9. Perform Insert, Update and Delete operations using SQL Server

LIST OF EXPERIMENTS/EXERCISES

5. C# Basic Programs
6. C# Programs on DataTypes & Operators
7. C# Programs on Numbers
8. C# Programs on Date, Time & Year
9. Special C# Programs
10. Conversion Programs in C#
11. Pattern Programs in C#
12. C# Programs on Math Functions
13. C# Programs on GCD, LCM & HCF
14. C# Programs on Bitwise Operations
15. C# Programs on Array
16. C# Programs on Matrix
17. C# Programs on Strings
18. C# Programs on Sorting
19. C# Programs on Preprocessor Attributes
20. Data Structures & Algorithms in C#
21. C# Programs on LINQ
22. C# Programs on Functions

23. C# Programs on Delegates
24. C# Programs on Inheritance
25. C# Programs on Operator Overloading
26. C# Programs on File Handling
27. C# Programs on Event Handling
28. C# Programs on Exception Handling

Text and Reference Book:

1. Web Programming Using ASP.NET (Hitesh Patel, Isha, Hemal) publication Atul prakashan
2. Tata McGraw Hill Education Private Limited (TMH)

Suggested E-resources.

1. NPTEL
2. Coursera

Scheme of EoSE :

(i)Exercise(s):45

(ii)Viva Voce: 20

(iii) Evaluation of record book: 15

BCA V SEMESTER

Code of the course : BCA7111T
Title of the course : Fundamentals of Machine Learning
Level of the Course : NHEQF Level 5.5
Credit of the Course : 4
Type of the Course : Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course : Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem solving.

Prerequisites : None

Objectives of the Course : To discover patterns in your data and then make predictions based on often complex findings to answer business questions, detect and analyze trends and help solve problems.

Learning Outcomes:

1. Understand a wide variety of learning algorithms.
2. Understand how to evaluate models generated from data.
3. Apply the algorithms to a real problem, optimize the models learned and report on the expected accuracy that can be achieved by applying the models.

Unit-I

Introduction: Introduction to Artificial Intelligence, Background and Applications, AI techniques, Tic Tac Toe problem, Problem Characteristics.

(12 Lectures)

Unit – II

Problem Solving and Searching Techniques: Problem Characteristics, Production Systems, Water Jug Problem, Control Strategies, Breadth First Search, Depth First Search, Hill climbing and its Variations, Heuristics Search Techniques: Best First Search. 8-Puzzle Problem.

(12 Lectures)

Unit – III

Knowledge Representation: Definition of Knowledge, Knowledge-Based Systems, Representation of Knowledge. Introduction to First-Order Predicate Logic, Conversion to clausal form, Unification, Resolution Principle.

(12 Lectures)

Unit – IV

Expert Systems: Introduction to Expert Systems, Characteristic Features of Expert Systems, Applications of Expert Systems, Components and Working of Expert Systems

(12 Lectures)

Unit – V

Introduction to Machine Learning Techniques: Fuzzy Logic, Fuzzy Set, Membership Function, Union, intersection, and complement of a fuzzy set, Introduction to Artificial Neural Network, Introduction to Support Vector Machine.

(12 Lectures)

Text and Reference Books:

1. DAN.W. Patterson, Introduction to A.I and Expert Systems – PHI, 2007.
2. Russell & Norvig, Artificial Intelligence - A Modern Approach, LPE, Pearson Prentice Hall, 2nd edition, 2005. (3) Rich & Knight, Artificial Intelligence – Tata McGraw Hill, 2nd edition, 1991.
3. W.F. Clocksin and Mellish, Programming in PROLOG, Narosa Publishing House, 3rd edition, 2001.

Suggested E-resources:

1. <https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-034-artificial-intelligence-fall-2010/lecture-videos/>
2. <https://nptel.ac.in/courses/106/102/106102220/>
3. <https://nptel.ac.in/courses/106/105/106105078/>

BCA V SEMESTER

Code of the course	: BCA7111P
Title of the course	: Machine Learning Lab
Level of the Course	: NHEQF Level 5.5
Credit of the Course	: 4
Type of the Course	: Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course	: Practical 60.
Prerequisites	: None

Course objectives :

1. Make use of Datasets in implementing the machine learning algorithms
2. Implement the machine learning concepts and algorithms in any suitable language of choice.

Learning outcomes:

1. Understand the implementation procedures for the machine learning algorithms.
2. Design Java/Python programs for various Learning algorithms.
3. Apply appropriate datasets to the Machine Learning algorithms.
4. Identify and apply Machine Learning algorithms to solve real-world problems. Conduction of Practical

Lab Experiments:

1. Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.
2. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.
3. Write a program to demonstrate the working of the decision tree-based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
4. Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate datasets.
5. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering a few test datasets.
6. Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your dataset.
7. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using the standard Heart Disease Data Set. You can use Java/Python ML library classes/API.
8. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.
9. Write a program to implement k-Nearest Neighbor algorithm to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem.
10. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate dataset for your experiment and draw graphs.

Text and Reference Books:

1. DAN.W. Patterson, Introduction to A.I and Expert Systems – PHI, 2007.
2. Russell & Norvig, Artificial Intelligence - A Modern Approach, LPE, Pearson Prentice Hall, 2nd edition, 2005. (3) Rich & Knight, Artificial Intelligence – Tata McGraw Hill, 2nd edition, 1991.

3. W.F. Clocksin and Mellish, Programming in PROLOG, Narosa Publishing House, 3rd edition, 2001.

Suggested E-resources:

1. <https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-034-artificial-intelligence-fall-2010/lecture-videos/>
2. <https://nptel.ac.in/courses/106/102/106102220/>
3. <https://nptel.ac.in/courses/106/105/106105078/>

Scheme of EoSE :

(i)Exercise(s):45

(ii)Viva Voce: 20

(iii) Evaluation of record book: 15

BCA V SEMESTER

Code of the course : BCA7112T
Title of the course : Software Engineering
Level of the Course : NHEQF Level 5.5
Credit of the Course : 4
Type of the Course : Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course : Lecture, 40+20=60. The 40 lectures for content delivery and 20 hours on diagnostic assessment, formative assessment, and subject/class activity, problem-solving.

Prerequisites : None

Course Objective :

1. Get better Understand, Analyze and Model User's Requirements, Select Appropriate Process Model Apply it to All Stages of Software Development Life Cycle (SDLC)
2. Select and Apply Appropriate Metrics to Estimate Software Size, Effort, and Cost
3. Understand the Characteristics and Applicability of Various Software Tools
4. Able to teach the students tried-and-tested techniques widely embraced by experienced analysts plus new and emerging tools and techniques.
5. Get a good balanced exposure to both traditional and object-oriented approaches to system analysis & design.

Learning Outcomes:

1. Be successful professionals in the field with solid fundamental knowledge of software engineering
2. Utilize and exhibit strong communication and interpersonal skills, as well as professional and ethical principles when functioning as members and leaders of multi-disciplinary teams
3. Apply their foundations in software engineering to adapt to readily changing environments using the appropriate theory, principles, and processes

Unit I

Introduction to Software Engineering & Process Models Software, Software Engineering, Software Characteristics, Software Application, Software Process, Layered Technology. Process Models – Waterfall, Incremental, Evolutionary Process Model – Prototype and Spiral Model.

(12 Lectures)

Unit II

Agile Process; Extreme Programming (XP), Brief Overview of Other Agile Process Models: Adaptive Software Development, Scrum Software Process and Project Metrics: Measures, Metrics and Indicators, Software measurement: Size-Oriented Metrics, Function -Oriented Metrics, Extended Function point metrics Software

(12 Lectures)

Unit III

Project Planning: Project Planning Objectives, Software Project Estimation, and Decomposition Techniques - Problem-Based Estimation, Process-Based Estimation, Empirical Estimation Models - The COCOMO Model Risk Analysis and Management: Software risks, Risk identification, Risk Projection, Risk Refinement, Risk Mitigation, Monitoring and Management.

(12 Lectures)

Unit IV

Software Quality Assurance: Basic concepts- Quality, Quality Control, Quality Assurance, Cost of Quality, Software Quality Assurance (SQA), Formal Technical Review Software Configuration Management: Baselines, Software Configuration Items, the SCM Process, Version Control, Change Control, Configuration Audit, Status Reporting Analysis Concepts and Principles: Requirements Elicitation for Software, Analysis Principles - The Information Domain, Modeling, Partitioning, Essential and

Implementation Views, Specification: Specification Principles, Representation, The Software Requirement Specification (SRS)

(12 Lectures)

Unit V

Introduction to Object-Oriented Analysis and Design Overview of UML, Conceptual Model of UML, Architecture, Software development lifecycle

(12 Lectures)

Text and References Book:

1. Roger S. Pressman, "Software Engineering – A Practitioner's Approach," 7th Edition, McGraw Hill Publications
2. Sommerville, "Software Engineering," 8th Edition, Pearson Education
3. Waman S. Jawadkar, "Software Engineering – Principles and Practices," TMGH Publications
4. Systems Analysis and Design by Kendall & Kendall, PHI Publication, 7th Edition
5. The Unified Modeling Language - User Guide by Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education Publication, 2009 Reprint
6. Object-Oriented Modeling and Design with UML by Michael Blaha, James Rumbaugh, Pearson Education Publication, 2nd Edition, 2007 Reprint
7. Object Oriented Analysis and Design Using UML by Mahesh P. Matha, PHI Publication
8. James Senn, Analysis and Design of Information system, McGraw Hill

Suggested E-resources:

1. Software Engineering | Software Quality Assurance - GeeksforGeeks
2. Software Engineering | Software Configuration Management - javatpoint
3. Analysis concepts and principles | PPT (slideshare.net)

BCA V SEMESTER

Code of the course	: BCA7112P
Title of the course	: Software engineering lab
Level of the Course	: NHEQF Level 5.5
Credit of the Course	: 2
Type of the Course	: Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course	: Practical 60.
Prerequisites	: A scripting language, IDEs, (integrated developments environment), Databases, system development lifecycle (SDLC)

Objectives of the Course:

1. To impart state-of-the-art knowledge on Software Engineering and UML in an interactive manner through the Web
2. Present case studies to demonstrate the practical applications of different concepts
3. Provide a scope to the students where they can solve small, real-life problems

Learning Outcomes:

1. Understand the role of software
2. Determine the problems occurred due to various software crisis.
3. Understand the need of requirements engineering process.
4. Compare the process of requirements development and requirements management.
5. Determine the importance of requirements classification.
6. Understand the difference between verification and validation process.
7. Determine the principle of design stating high cohesion and low coupling.
8. Determine the procedure of regression testing.
9. Understand the importance of performance testing.
10. Determine the concepts of software metrics used before software deployment.

List of experiments:

1. An introduction to software engineering.
2. Identifying the Requirements from Problem Statements
3. Estimation of Project Metrics
4. Development of DFD, data dictionary, E-R diagram, structured chart for the project.
5. To study and draw various UML diagrams.
6. To illustrate the use of class diagrams.
7. To draw an activity diagram and use case diagram for ATM and Library Management System.
8. Modeling Data Flow Diagrams
9. Estimation of Test Coverage Metrics and Structural Complexity
10. Draw Object Diagram for ATM System.
11. Identifying Domain Classes from the Problem Statements
12. Draw ER Diagram for Hospital Management System.

Text and References books

1. K.K. Aggarwal & Yogesh Singh, "Software Engineering," New Age International, 2005
2. Pankaj Jalote, "An Integrated Approach to Software Engineering," Second Edition, Springer.

Suggested E-resources:

1. <http://vlabs.iitkgp.ernet.in/se/>
2. <https://www.ahirlabs.com/practicals/software-engineering-practical/>
3. <https://sites.google.com/view/ait-se/Home/assignment>

Scheme of EoSE :

- (i) Exercise(s): 45 (ii) Viva Voce: 20 (iii) Evaluation of record book: 15

Skill Enhancement Course (SEC)

BCA V SEMESTER

Course Code : SEI7333P
Title of the Course : R Programming
Level of the Course : NHEQF Level 5.5
Credit of the Course : 2
Type of the Course : Skill Enhancement Course (SEC) Course in computer science
Delivery Type of the Course : Practicum (20+10). 20hrs for lecture content delivery +10 hrs for diagnostic assessment, formative assessment, and subject/class activity, problem solving.

Prerequisites :

1. Must know of the basics of any programming language.
2. Should have a deep knowledge of statistics and mathematics.
3. Must have a good understanding of graphs and plots that are used for data visualization.

Objectives:

1. Students will develop relevant programming abilities.
2. Students will demonstrate proficiency with statistical analysis of data.
3. Students will develop the ability to build and assess data-based models.

Learning Outcomes:

1. Learn to create own customized functions
2. Able to construct tables and figures for descriptive statistics
3. Learn to understand new data sets and functions
4. Clear understanding on how to organize data and analyze data using real-time examples

UNIT I

Introduction to R: What is R? – Why R? – Advantages of R over Other Programming Languages - R Studio: R command Prompt, R script file, comments – Handling Packages in R: Installing an R Package, Few commands to get started: installed.packages(), packageDescription(), help(), find.package(), library() - Input and Output – Entering Data from keyboard – Printing fewer digits or more digits – Special Values functions: NA, Inf and -inf.

(12 Lectures)

UNIT II

R Data Types: Vectors, Lists, Matrices, Arrays, Factors, Data Frame – R - Variables: Variable assignment, Data types of Variable, Finding Variable ls(), Deleting Variables - R Operators: Arithmetic Operators, Relational Operators, Logical Operator, Assignment Operators, Miscellaneous Operators - R Decision Making: if statement, if – else statement, if – else if statement, switch statement – R Loops: repeat loop, while loop, for loop - Loop control statement: break statement, next statement

(12 Lectures)

UNIT III

R-Function: function definition, Built-in functions: mean(), paste(), sum(), min(), max(), seq(), user-defined function, calling a function, calling a function without an argument, calling a function with argument values - R-Strings – Manipulating Text in Data, R Vectors – Sequence vector, rep function, vector access, vector names, vector math, vector recycling, vector element sorting - R List - Creating a List, List Tags and Values, Add/Delete Element to or from a List, Size of List, Merging Lists, Converting List to Vector - R Matrices – Accessing Elements of a Matrix, Matrix Computations: Addition, subtraction, Multiplication and Division- R Arrays: Naming Columns and Rows, Accessing Array Elements, Manipulating Array Elements

(12 Lectures)

UNIT IV

Data Frames – Create Data Frame, Data Frame Access, Understanding Data in Data Frames: dim(), nrow(), ncol(), str(), Summary(), names(), head(), tail(), edit() functions - Extract Data from Data Frame, Expand Data Frame: Add Column, Add Row - Joining columns and rows in a Data frame rbind() and cbind() – Merging Data frames merge() – Melting and Casting data melt(), cast(). (3 Lectures) Loading and handling Data in R: Getting and Setting the Working Directory – getwd(), setwd(), dir() - R-CSV Files - Input as a CSV file, Reading a CSV File, Analyzing the CSV File: summary(), min(), max(), range(), mean(), median(), apply() - Writing into a CSV File – R - Excel File – Reading the Excel file.

(12 Lectures)

UNIT V

Descriptive Statistics: Data Range, Frequencies, Mode, Mean and Median: Mean Applying Trim Option, Applying NA Option, Median - Mode - Standard Deviation – Correlation - Spotting Problems in Data with Visualization: visually Checking Distributions for a single Variable - R – Pie Charts: Pie Chart title and Colors – Slice Percentages and Chart Legend, 3D Pie Chart – R Histograms – Density Plot - R – Bar Charts: Bar Chart Labels, Title and Colors.

(12 Lectures)

Text and Reference Books:

1. Sandip Rakshit, R Programming for Beginners, McGraw Hill Education (India), 2017
2. Seema Acharya, Data Analytics using R, McGrawHill Education (India), 2018.
3. Andrie de Vries, Joris Meys, R for Dummies A Wiley Brand, 2nd Edition, John Wiley and Sons, Inc, 2015

Suggested E-Resources:

1. R Programming, Retrieved from https://www.tutorialspoint.com/r/r_tutorial.pdf.

BCA V SEMESTER

Course Code : SEI7334P
Title of the Course : Introduction to MATLAB Level of the Course: NHEQF Level 5.5
Credit of the Course : 2
Type of the Course : Skill Enhancement Course (SEC) Course in computer science
Delivery Type of the Course : Practicum (20+10). 20hrs for lecture content delivery +10 hrs for diagnostic assessment, formative assessment, and subject/class activity, problem solving.

Prerequisites:

1. C Programming
2. Basics of Mathematics
3. Basic computer literacy is expected.

Objectives of the Course:

1. Programming knowledge in Research and Development
2. Data analysis, visualization, and programming
3. Scientific problem-solving with applications and examples

Learning Outcomes:

1. MATLAB combines nicely calculation and graphic plotting.
2. Relatively fast when performing matrix operations

Problem based on:-

1. Basic Commands
2. Arithmetic Operations
3. Complex numbers
4. Array
5. Working with Matrices
6. Expressions a. Variable b. Numbers c. Operators d. Functions
7. Relational and Logical Operations
8. Plotting Function
9. Complex and Statistical Functions (
10. Programming in MATLAB (M-files)
11. Input / Output of Variables (Numbers and Strings)
12. Special characters and functions (13) Flow Control a. if statements b. switch statements c. for loops d. while loops e. break statements
13. Plotting

Text and Reference Books:

1. The MathWorks Inc. MATLAB 7.0 (R14SP2). The MathWorks Inc., 2005.
2. S. J. Chapman. MATLAB Programming for Engineers. Thomson, 2004.
3. C. B. Moler. Numerical Computing with MATLAB. Siam, 2004.
4. D. J. Higham and N. J. Higham. MATLAB Guide. Siam, second edition, 2005.

Suggested E-Resources:

1. <https://www.mccormick.northwestern.edu/documents/students/undergraduate/introduction-to-matlab.pdf>
2. <https://uotechnology.edu.iq/dep/coe/lectures/dr-uosra/New%20folder/Matlab%20LAB2.pdf>
3. <https://ecs.utdallas.edu/loizou/ee3102/lab1.pdf>
4. <https://www.mathworks.com/content/dam/mathworks/mathworks-dot-com/moler/exm/book.pdf>
5. <http://srmcem.ac.in/pdf/KEC-453%20S&S%20Lab%20manual.pdf>

BCA V SEMESTER

Code of the course	: SEI7335P
Title of the course	: Digital Marketing Lab
Level of the Course	: NHEQF Level 5.5
Credit of the Course	: 2
Type of the Course	: Skill Enhancement Course (SEC) Course in computer science
Delivery Type of the Course	: Practicum (20+10). 20 hrs for lecture content delivery +10 hrs for diagnostic assessment, formative assessment, and subject/class activity, problem solving.
Prerequisites	: None

Objectives of the Course:

The course allows us to make students gain practical knowledge of digital marketing using google, facebook, twitter and mails etc.

Learning Outcomes:

1. Understand the importance of digital marketing in a rapidly changing business landscape
2. Learn the key elements of a digital marketing strategy
3. Examine how effectiveness of a digital marketing campaign can be measured.
4. Demonstrate knowledge and ability to implement common digital marketing tools such as SEO, SEM, Social media and Blogs.

Syllabus List of Experiments:

1. Digital Marketing Implementation in Business Scenario
2. Create the Digital Marketing Webpage
3. Conducting the Search Engine Optimization and Search Engine Marketing
4. Using Google Analytics to analyze website performance
5. Creating Promotional banner through Canva
6. Facebook Promotion using banners
7. Creating YouTube Channel for Marketing
8. Twitter Marketing
9. Instagram Marketing
10. Email Marketing

Text and Reference Books:

1. Ian Dodson, "The Art of Digital Marketing", Wiley 2018.
2. Kotler, "Marketing 4.0: Moving from Traditional to Digital", Wiley, 2017
3. Seema Gupta, "Digital Marketing", Mc-Graw Hill, First Edition, 2017

Suggested E-resources:

1. A step-by-step guide to structuring a digital marketing plan in 2020 (wearemarketing.com)
2. How to Create a Twitter Marketing Strategy | Sprout Social

Scheme of EoSE :

(i)Exercise(s):45

(ii)Viva Voce: 20

(iii) Evaluation of record book: 15

BCA V SEMESTER

Code of the Course	: SEI7336P
Title of the Course	: Open source Programming Lab
Level of the Course	: NHEQF Level 5.5
Credit of the Course	: 2
Type of the Course	: Skill Enhancement Course (SEC) Course in computer science
Delivery Type of the Course	: Practicum (20+10). 20hrs for lecture content delivery +10 hrs for diagnostic assessment, formative assessment, and subject/class activity, problem solving.

Prerequisites:

1. one must have a basic understanding of computer programming, Internet, database, HTML/XHTML and MySQL will be very helpful.
2. It is designed for those who are unaware of the PHP concepts but have a basic understanding of computer programming.

Objectives of the Course:

1. Analyze the basic structure of a PHP web application and be able to install and maintain the web server, compile, and run a simple web application.
2. Learn how databases work and how to design one, as well as how to use PHPMyAdmin to work with MySQL.

Learning Outcomes:

1. List the major elements of the PHP & MySQL work and explain why PHP is good for web development
2. Learn how to take a static website and run from a database using PHP and MySQL.
3. Analyze the basic structure of a PHP web application and be able to install and maintain the web server, compile, and run a simple web application.
4. Learn how databases work and how to design one, as well as how to use PHP MyAdmin to work with MySQL.
5. Learn different ways of connecting to MySQL through PHP, and how to create tables, enter data, select data, change data, and delete data. Connect to SQL Server and other data sources.

List of Experiments:

1. Write a PHP program to print sum of digits.
2. Write a PHP program to print factorial of a number.
3. Write a program to print "Hello World" using echo only?
4. Write a program to print "Hello PHP" using php variable?
5. Write a Program to display count, from 5 to 15 using PHP loop
6. Write a PHP program to check armstrong number.
7. Write a PHP program to check palindrome number.
8. Write a program to check student grade based on the marks
9. Write a program to show day of the week (for example: Monday) based on numbers using switch/case statements.
10. Write a PHP program using nested for loop that creates a chess board.
11. Write a PHP program to calculate electricity bill using if-else conditions.
12. Write a PHP program to Array manipulation.
13. write a simple calculator program in PHP using switch case.
14. write a program in PHP to remove specific element by value from an array using PHP program.
15. Create a PHP page for login page with sql connection.
16. Write a PHP program to Read from existing file.
17. Write a PHP program to Write a file
18. Write a PHP program to calculate Date and Time function.
19. Get name of the user from a form and show greeting text.

20. Write a PHP program to design Curriculum Vitae.
21. Write a PHP program hit counter using cookies.
22. Create a web page to advertise a product of the company using images and audio.
23. Write a PHP script for the following: Design a form to accept the details of 5 different items, such as item code, item name, units sold, rate. Display the bill in the tabular format. Use only 4 text boxes. (Hint : Use of explode function.)
24. Create a form to accept student information (name, clas, address). Once the student information is accepted, accept marks in next form (Phy, Bio, Chem, Maths, Marathi, English) .Display the mark sheet for the student in the next form containing name, class, marks of the subject, total and percentage.
25. Create a form to accept customer information(name, address, ph-no).Once the customer information is accepted, accept product information in the next form(Product name, qty, rate). Display the bill for the customer in the next form. Bill should contain the customer information and the information of the products entered.
26. Create a web page for Travel agency.
27. Create a web page for software company websites.
28. Create a PHP page for login system using session.

Text and References Books:

1. The Joy of PHP Programming: A Beginner's guide – by Alan Forbes.
2. PHP For Web by Larry Ullman.
3. Head First PHP & MySQL: A Brain-Friendly Guide.

Suggested E-resources.

1. <https://www.simplilearn.com/tutorials/PHP-tutorial/PHP-login-form>
2. <https://www.youtube.com/watch?v=aIsu9SPcGbU>
3. <https://PHPpot.com/PHP/user-registration-in-PHP-with-login-form-with-mysql-and-code-download/>
4. <https://www.netsolutions.com/insights/what-is-PHP/#:~:text=PHP%20is%20a%20server%2Dside,in%20the%20web%20development%20world.>

Scheme of EoSE :

(i)Exercise(s):45

(ii)Viva Voce: 20

(iii) Evaluation of record book: 15

BCA V SEMESTER

Code of the course : SEI7337P
Title of the course : Big Data Lab
Level of the Course : NHEQF Level 5.5
Credit of the Course : 2
Type of the Course : Skill Enhancement Course (SEC) Course in computer science
Delivery Type of the Course : Practicum (20+10). 20hrs for lecture content delivery +10 hrs for diagnostic assessment, formative assessment, and subject/class activity, problem solving.

Prerequisites:

1. SQL, Data Warehousing/Data Processing, and Database Knowledge: This includes SQL knowledge to query data and manipulate information stored in databases.
2. Java, Scala, and Python Programming are the essential languages in the data analytics domain.

Objectives of the Course:

1. Big data enables you to gather data from social media, web visits, call logs
2. To improve the interaction experience and maximize the value delivered.

Learning Outcomes:

1. Preparing for data summarization, query, and analysis.
2. Applying data modeling techniques to large data sets.
3. Creating applications for Big Data analytics.
4. Building a complete business data analytic solution.

List of Experiments:

1. Downloading and installing Hadoop; Understanding different Hadoop modes. Startup scripts, Configuration files.
2. Implement the following file management tasks in Hadoop:
 - a. Adding files and directories
 - b. Retrieving files
3. Deleting files Hint: A typical Hadoop workflow creates data files (such as log files) elsewhere and copies them into HDFS using one of the above command line utilities.
4. Implement of Matrix Multiplication with Hadoop Map Reduce.
5. Write a Map Reduce program that mines weather data. Hint: Weather sensors collecting data every hour at many locations across the globe gather a large volume of log data, which is a good candidate for analysis with Map Reduce, since it is semi-structured and record-oriented.
6. Run a basic Word Count Map Reduce program to understand Map Reduce Paradigm.
7. Implementation of K-means clustering using Map Reduce.
8. Patrice importing and exporting data from various databases.
9. Write PIG Commands: Write Pig Latin scripts sort, group, join, project, and filter your data.
10. Run the Pig Latin Scripts to find Word Count
11. Run the Pig Latin Scripts to find a max temp for each and every year.

Text and Reference Books:

1. Big Data Analytics Lab Manual: Step by Step Guide to Hadoop, Pig, Hive, and MongoDB Kindle Edition by Dr. M.S.Vijaya Dr. N.Radha V. Pream Sudha, Dr. N. Radha Narayanan

Suggested E-resources:

1. [BIG DATA ANALYTICS Lab Manual.pdf (mrcet.com)](<https://www.example.com>)
2. [K-Means Clustering Algorithm - Javatpoint](<https://www.example.com>)

Scheme of EoSE :

(i)Exercise(s):45 (ii)Viva Voce: 20 (iii) Evaluation of record book: 15

Skill Enhancement Course (SEC)

Code of the course : SEI6338P
Title of the course : Data Mining and Knowledge Management LAB
Level of the Course : NHEQF Level 5.5
Credit of the Course : 2
Type of the Course : Skill Enhancement Course (SEC) Course in computer science
Delivery Type of the Course : Practicum (20+10). 20hrs for lecture content delivery +10 hrs for diagnostic assessment, formative assessment, and subject/class activity, problem solving.

Prerequisites : Basic Knowledge of Data Mining and Weka.

Objectives of the Course:

1. Learn how to build a data warehouse and query it (using open source tools like Pentaho Data Integration Tool, Pentaho Business Analytics).
2. Learn to perform data mining tasks using a data mining toolkit (such as open source WEKA).
3. Understand the data sets and data preprocessing.
4. Demonstrate the working of algorithms for data mining tasks such as association rule mining, classification, clustering, and regression.
5. Exercise the data mining techniques with varied input values for different parameters.
6. To obtain Practical Experience Working with all real data sets.

Learning Outcomes:

1. Ability to add mining algorithms as a component to the existing tools
2. Demonstrate the classification, clustering and etc. in large data sets.
3. Ability to apply mining techniques for realistic data

Syllabus List of Experiments:

1. Installation of WEKA Tool
2. Creating a new Arff File
3. Pre-Processes Techniques on Data Set: Pre-process a given dataset based on Handling Missing Values
4. Generate Association Rules using the Apriori Algorithm
5. Generating association rules using fp growth algorithm
6. Build a Decision Tree by using J48 algorithm
7. Naïve bayes classification on a given data set
8. Applying k-means clustering on a given data set
9. Calculating Information gains measures
10. OLAP Cube and its different operations
11. Case Study on various Knowledge Management Tools.
12. Case Study on knowledge application at the Individual level & the Group and Organizational level

Text and Reference Books:

1. Jiawei Han and Micheline Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers, third edition, 2013
2. Pang-Ning Tan, Michael Steinbach, Anuj Karpatne, Vipin Kumar, Introduction to Data Mining, second edition, Pearson, 2019
3. Ian.H.Witten, Eibe Frank and Mark.A.Hall, Data Mining: Practical Machine Learning Tools and Techniques, third edition, 2017

Suggested E-resources.

1. [NPTEL](<https://www.example.com>)
2. [Coursera](<https://www.example.com>)

Scheme of EoSE :

(i)Exercise(s):45 (ii)Viva Voce: 20 (iii) Evaluation of record book: 1

BCA VI SEMESTER

Code of the course : BCA711XP
Title of the course : Project
Level of the Course : NHEQF Level 5.5
Credit of the Course : 20
Type of the Course : Discipline Specific Elective (DSE) Course in Computer Science
Delivery Type of the Course : Practical 300

Objectives of the Course:

In house project must be done by each student on simple applications using any computer language/ RDBMS/ Web design/visual programming etc. The total work must be of minimum 180 hours per student. The internal guide must schedule the work & evaluate internally from time to time. The project report must be prepared for the external examination. Monthly report of the students must be taken to monitor progress and must be placed for evaluation by external examiner. Projects submitted by the students shall be evaluated during external evaluation to ensure independent contribution and proficiency acquired by the students. Note: Students must be allotted projects in the beginning of the session. Candidates submitting readymade projects/copied/ projects developed by professionals in the market etc shall be awarded zero marks. Two copies of the project report and the software developed must be submitted to the external examiner. One copy of the project shall be returned to the student with the signature of external examiner.

BCA VI SEMESTER

Code of the course : SEI7339S
Title of the course : Report Writing
Level of the Course : NHEQF Level 5.5
Credit of the Course : 2
Type of the Course : Skill Enhancement Course (SEC) Course in computer science
Delivery Type of the Course : 30

Objectives of the Course:

Research report based on topic selected by student based on guidance of faculty/industry expert/any other mechanism of delivery adopted by department committee at the execution of course etc.

The total work must be of minimum 30 hours per student. The internal guide will schedule the work & evaluate internally from time to time. The report must be prepared for the external examination. report of the students must be taken to monitor progress and must be placed for evaluation by external examiner. Report submitted by the students shall be evaluated during external evaluation to ensure independent contribution and proficiency acquired by the students.

Note: Students must be allotted topic in the beginning of the session. Candidates submitting readymade report of projects/copied projects developed by professionals in the market etc shall be awarded zero marks. Two copies of the report must be submitted to the external examiner. One copy of the project shall be returned to the student with the signature of examiner.