Syllabus For Four Years B.Sc. Computer Science (Minor)

under

National Education Policy (NEP) Framework

w.e.f Academic Year (2024)

Course Type	Course Code	Course Title	Credit	L-T-P	Marks		
					CA	ESE	TOTAL
Minor-1	**COSMI-1**	T: Computer fundamentals & C programming P: Programming in C Lab	4	3-0-1	15	60	75
Minor-2	**COSMI-2**	Data Structures Data Structures Lab	4	3-0-1	15	60	75
Minor -3	**COSMI-3**	Introductions to DBMS DBMS Lab	4	3-0-1	15	60	75
Minor -4	**COSMI-4**	T: Introduction To Data Science with python Introduction To Data Science Lab	4	3-0-1	15	60	75

COSMI-1:
Computer Fundamentals and C programming (40 Lectures)

Course outcome:

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

1. Converse in basic computer terminology
2. Formulate opinions about the impact of computers on society
3. Possess the knowledge of basic hardware peripherals
4. Know and use different number systems and the basics of programming
5. Solve basic computational problems with C language

Unit-I

Module I: Introduction to Computer systems (3 Lectures)

Generations of Computer (I-V), number system(Binary, Decimal, Hexadecimal ,Octal Numbers, Signed Numbers), Block Diagram of a Computer, Functions of the Different Units Input unit ,Output unit, Memory unit, CPU (ALU+CU).

Module II: Input & Output Devices (2 Lectures)

Input Devices: Keyboard, Point and draw devices, mouse, joystick, track ball, light pen , Data Scanning devices image scanner, OCR, OMR, MICR, Bar code reader, card reader, Voice Recognition Device, Digitizers Output Devices: Monitor, Printer laser printer, dot matrix printer, ink jet printer , Projector

Module III:.Memories (Memory hierarchy) (3 Lectures)

Registers (Types of Registers),Cache Memory, Primary Memory , RAM : How data is stored in a RAM, DRAM and SRAM, ROM: ROM BIOS/ Firmware, Types of ROM Secondary Memories: Hard disk : Structure of a hard disk, how data is stored in a hard disk, concept of tracks, sectors, clusters, cylinders, formatting of hard disk (low level formatting and high level formatting) Floppy (data storage mechanism) CD (data storage mechanism).

Module IV: System Software (4 Lectures)

Operating System , Functions of OS, Types of OS, Program Language Translators : Assembler, Compiler, Interpreter. Utility Programs, Communication Software, Performance Monitoring Software, Application Software. Software hierarchy and dependence between the different layers.

Computer Languages: Machine language, Assembly language, High level language.

Unit –II: Programming In C

Module I:

Introduction to Programming (3 Lectures)

Problem analysis, Flowchart, algorithms, Pseudo codes, structured programming, Example of Flowchart and Algorithm representation, Brief History of Development of C language, Features of C language, Process of compiling and running a C program, Overview of Procedural Programming, Performance Analysis and Measurements: Time and Space complexity , Use of main() function.

Module II: Data Types, Variables, Constants, Operators and Basic I/O (4 Lectures)

Declaring, Defining and Initializing Variables, Scope of Variables, Using Named Constants, Keywords,Data Types, Casting of Data Types, Operators (Arithmetic, Logical and Bitwise), Using Comments in programs, Character I/O (getc, getchar, putc, putchar etc), Formatted and Console I/O (printf(), scanf()),using basic header files (stdio.h, conio.h etc).

Module III: Expressions, Conditional Statements and Iterative Statements (4 Lectures)

Simple Expressions in C (including Unary Operator Expressions, Binary Operator Expressions), Understanding Operators, Precedence in Expressions, Conditional Statements (if construct, switchcase construct), Understanding syntax and utility of Iterative Statements (while, do-while, and for loops), Use of break and continue in Loops, Using Nested Statements (Conditional as well as Iterative)

Module IV: Functions and Arrays (6 Lectures)

Utility of functions, Call by Value, Call by address, Functions returning value, Void functions, Return data type of functions, Functions parameters, Differentiating between Declaration and Definition of Functions, Recursion. Creating and Using One Dimensional Arrays (Declaring and Defining an Array, Initializing an Array, Accessing individual elements in an Array, Manipulating array elements using loops), Use Various types of arrays (integer, float and character arrays / Strings) Two- dimensional Arrays (Declaring, Defining and Initializing Two Dimensional Array, Working with Rows and Columns), Introduction to Multi-dimensional arrays. Different storage classes and scope of variables.

Module V: Derived Data Types (Structures and Unions) (2 Lectures)

Understanding utility of structures and unions, Declaring, initializing and using simple structures and unions, Manipulating individual members of structures and unions, Array of Structures, Individual datamembers as structures, Passing and returning structures from functions, Structure with union as members, Union with structures as members, enum.

Module VI: String (1 Lectures)

Character Array, Library functions related to string (strcat(), strcmp(), strcpy(), strlen())

Module VI:Pointers (2 Lectures)

Understanding a Pointer Variable, Simple use of Pointers (Declaring and Dereferencing Pointers to simple variables), Pointers to Pointers, Pointers to structures, Problems with Pointers, Passing pointers as function arguments, Returning a pointer from a function, using arrays as pointers, Passing arrays to functions. Pointer arithmetic. Command Line Arguments.

Module VI: Dynamic Memory Allocation (3 Lectures)

Differentiating between static and dynamic memory allocation, use of malloc, calloc, realloc and free functions, storage of variables in static and dynamic memory allocation

Module VI:File I/O, Preprocessor Directives (3 Lectures)

Opening and closing a file (use of fopen, fclose), Reading and writing Text Files, using fscanf(),fprintf(), read() and write() functions, Random access in files, Understanding the Preprocessor Directives (#include, #define, #error, #if, #else, #elif, #endif, #ifdef, #ifndef and #undef), Macros.

Recommended Books

1. Dey, Bandyopadhyay, C Programming Essentials, Pearson Education.

2. Balagurusamy, Programming in ANSI C, Tata McGraw Hill.

3. Kerninghan, Ritchie, The C Programming Language, Pearson.

4. Kanetkar, Let Us C, BPB Publications

C Programming Lab

1. WAP to print the sum and product of digits of an integer.
2. WAP to reverse a number.
3. WAP to compute the sum of the first n terms of the following series S = 1+1/2+1/3+1/4+……
4. WAP to compute the sum of the first n terms of the following series

   S =1-2+3-4+5…………….

5. Write a function that checks whether a given string is Palindrome or not. Use this function to find whether the string entered by user is Palindrome or not.

6. Write a function to find whether a given no. is prime or not. Use the same to generate the prime numbers less than 100.

7. WAP to compute the factors of a given number.

8. Write a macro that swaps two numbers. WAP to use it.

9. WAP to print a triangle of stars as follows (take number of lines from user):

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

10. WAP to perform following actions on an array entered by the user:
    i) Print the even-valued elements
    ii) Print the odd-valued elements
    iii) Calculate and print the sum and average of the elements of array

iv) Print the maximum and minimum element of array
v) Remove the duplicates from the array
vi) Print the array in reverse order

The program should present a menu to the user and ask for one of the options. The menu should also include options to re-enter array and to quit the program.

11. WAP that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.

12. Write a program that swaps two numbers using pointers.

13. Write a program in which a function is passed address of two variables and then alter its contents.

14. Write a program which takes the radius of a circle as input from the user, passes it to another function that computes the area and the circumference of the circle and displays the value of area and circumference from the main() function.

15. Write a program to find sum of n elements entered by the user. To write this program, allocate memory dynamically using malloc() / calloc() functions or new operator.

16. Write a menu driven program to perform following operations on strings:

    a) Show address of each character in string
    b) Concatenate two strings without using strcat function.
    c) Concatenate two strings using strcat function.
    d) Compare two strings
    e) Calculate length of the string (use pointers)
    f) Convert all lowercase characters to uppercase
    g) Convert all uppercase characters to lowercase
    h) Calculate number of vowels
    i) Reverse the string

17. Given two ordered arrays of integers, write a program to merge the two-arrays to get an ordered array.

18. WAP to display Fibonacci series (i)using recursion, (ii) using iteration

19. WAP to calculate Factorial of a number (i)using recursion, (ii) using iteration

20. WAP to calculate GCD of two numbers (i) with recursion (ii) without recursion.

21. Create Matrix class using templates. Write a menu-driven program to perform following Matrix
    operations (2-D array implementation):
    a) Sum b) Difference c) Product d) Transpose

22. Create the Person class. Create some objects of this class (by taking information from the user). Inherit the class Person to create two classes Teacher and Student class. Maintain the respective information in the classes and create, display and delete objects of these two classes (Use Runtime Polymorphism).

23. Create a class Triangle. Include overloaded functions for calculating area.

Overload assignment operator and equality operator.

24. Create a class Box containing length, breath and height. Include following methods in it:
    a) Calculate surface Area
    b) Calculate Volume
    c) Increment, Overload ++ operator (both prefix & postfix)
    d) Decrement, Overload -- operator (both prefix & postfix)
    e) Overload operator == (to check equality of two boxes), as a friend function
    f) Overload Assignment operator
    g) Check if it is a Cube or cuboid

    Write a program which takes input from the user for length, breath and height to test the above class.

25. Create a structure Student containing fields for Roll No., Name, Class, Year and Total Marks. Create 10 students and store them in a file.

26. Write a program to retrieve the student information from file created in previous question and print it in following format:

    Roll No. Name Marks

27. Copy the contents of one text file to another file, after removing all whitespaces.

28. Write a function that reverses the elements of an array in place. The function must accept only one pointer value and return void.

29. Write a program that will read 10 integers from user and store them in an array. Implement array using pointers. The program will print the array elements in ascending and descending order.

Recommended Books

1. Dey, Bandyopadhyay, C Programming Essentials, Pearson Education.

2. Balagurusamy, Programming in ANSI C, Tata McGraw Hill.

3. Kerninghan, Ritchie, The C Programming Language, Pearson.

4. Kanetkar, Let Us C, BPB Publications.

COSMI-2:
Data Structures (40 Lectures)

Course Outcome

Upon Completing the Course, Students will able to:

1. Learn the basic types for data structure, implementation and application.

2. Know the strength and weakness of different data structures.

3. Use the appropriate data structure in context of solution of given problem..

4. Develop programming skills which require to solve given problem

1.Arrays **(5 Lectures)**

Single and Multi-dimensional Arrays, Sparse Matrices (Array and Linked Representation)

2.Stacks **(5 Lectures)**

Implementing single / multiple stack/s in an Array; Prefix, Infix and Postfix expressions, Utility and conversion of these expressions from one to another; Applications of stack;Limitations of Array representation of stack

3.Linked Lists **(6 Lectures)**

Singly, Doubly and Circular Lists (Array and Linked representation);Normal and Circular representation of Stack in Lists; Self Organizing Lists; Skip Lists

4.Queues **(5 Lectures)**

Array and Linked representation of Queue, De-queue, Priority Queues

5.Recursion **(5 lectures)**

Developing Recursive Definition of Simple Problems and their implementation; Advantages and Limitations of Recursion; Understanding what goes behind Recursion (Internal Stack Implementation)

6.Trees **(8 Lectures)**

Introduction to Tree as a data structure; Binary Trees (Insertion, Deletion , Recursive and Iterative Traversals on Binary Search Trees); Threaded Binary Trees (Insertion, Deletion, Traversals); Height-Balanced Trees (Various operations on AVL Trees).

7.Searching and Sorting **(3 Lectures)**

Linear Search, Binary Search, Comparison of Linear and Binary Search, Selection Sort,Insertion Sort, Insertion Sort, Shell Sort, Comparison of Sorting Techniques

8.Hashing **(3 Lectures)**

Introduction to Hashing, Deleting from Hash Table, Efficiency of Rehash Methods, Hash Table Reordering, Resolving collusion by Open Addressing, Coalesced Hashing, Separate Chaining, Dynamic and Extendible Hashing, Choosing a Hash Function, Perfect Hashing Function

Reference Books:

1.Adam Drozdek, "Data Structures and algorithm in C++", Third Edition, Cengage Learning, 2012.

2.SartajSahni, Data Structures, "Algorithms and applications in C++", Second Edition, Universities Press, 2011.

3.Aaron M. Tenenbaum, Moshe J. Augenstein, YedidyahLangsam, "Data Structures Using C and C++:, Second edition, PHI, 2009.

4.Robert L. Kruse, "Data Structures and Program Design in C++", Pearson,1999.

5.D.S Malik, Data Structure using C++,Second edition, Cengage Learning, 2010.

6.Mark Allen Weiss, "Data Structures and Algorithms Analysis in Java", Pearson Education, 3rd edition, 2011

7.Aaron M. Tenenbaum, Moshe J. Augenstein, YedidyahLangsam, "Data Structures Using Java, 2003.

8. Robert Lafore, "Data Structures and Algorithms in Java, 2/E", Pearson/ Macmillan Computer Pub,2003

9. John Hubbard, "Data Structures with JAVA", McGraw Hill Education (India) Private Limited; 2 edition, 2009

10. Goodrich, M. and Tamassia, R. "Data Structures and Algorithms Analysis in Java", 4th Edition, Wiley,2013

11. Herbert Schildt, "Java The Complete Reference (English) 9th Edition Paperback", Tata McGraw Hill, 2014.

12. D. S. Malik, P.S. Nair, "Data Structures Using Java", Course Technology, 2003.

Data Structures Lab: 20 Lectures

1.Write a program to search an element from a list. Give user the option to perform Linear or Binary search. Use Template functions.

2.WAP using templates to sort a list of elements. Give user the option to perform sorting using Insertion sort, Bubble sort or Selection sort.

3.Implement Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list and concatenate two linked lists (include a function and also overload operator +).

4.Implement Doubly Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.

5.Implement Circular Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.

6.Perform Stack operations using Linked List implementation.

7.Perform Stack operations using Array implementation. Use Templates.

8.Perform Queues operations using Circular Array implementation. Use Templates.

9.Create and perform different operations on Double-ended Queues using Linked List implementation.

10.WAP to scan a polynomial using linked list and add two polynomial.

11.WAP to calculate factorial and to compute the factors of a given no. (i)using recursion,

(ii) using iteration

12. (ii) WAP to display fibonacci series (i)using recursion, (ii) using iteration

13. WAP to calculate GCD of 2 number (i) with recursion (ii) without recursion

14. WAP to create a Binary Search Tree and include following operations in tree:

(a) Insertion (Recursive and Iterative Implementation)

(b) Deletion by copying

(c) Deletion by Merging

(d) Search a no. in BST

(e) Display its preorder, postorder and inorder traversals Recursively

(f) Display its preorder, postorder and inorder traversals Iteratively

(g) Display its level-by-level traversals

(h) Count the non-leaf nodes and leaf nodes

(i) Display height of tree

(j) Create a mirror image of tree

(k) Check whether two BSTs are equal or not

15. WAP to convert the Sparse Matrix into non-zero form and vice-versa.

16. WAP to reverse the order of the elements in the stack using additional stack.

17. WAP to reverse the order of the elements in the stack using additional Queue.

18. WAP to implement Diagonal Matrix using one-dimensional array.

19. WAP to implement Lower Triangular Matrix using one-dimensional array.

20. WAP to implement Upper Triangular Matrix using one-dimensional array.

21. WAP to implement Symmetric Matrix using one-dimensional array.

22. WAP to create a Threaded Binary Tree as per inorder traversal, and implement operations like finding the successor / predecessor of an element, insert an element, inorder traversal.

23. WAP to implement various operations on AVL Tree.

COSMI-3:

Database Management Systems Theory

Course Outcome:

Upon successful completion of the course, the student will be able to

1. Understand the basic concepts of database management systems
2. Apply SQL to find solutions to a broad range of queries.
3. Apply normalization techniques to improve database design.
4. Analyze a given database application scenario to use ER model for conceptual design of the database

1.Introduction (4 Lectures)

Characteristics of database approach, data models, database system architecture and data independence.

2.Entity Relationship(ER) Modeling (4 Lectures)

Entity types, relationships, constraints.

3.Relation data model (10 Lectures)

Relational model concepts, relational constraints, relational algebra, SQLqueries

4.Database design (5 Lectures)

Mapping ER/EER model to relational database, functional dependencies,Lossless decomposition, Normalforms(upto BCNF).

5.Transaction Processing (3 Lectures)

ACID properties, concurrency control

6.File Structure and Indexing (4 Lectures)

Operations on files, File of Unordered and ordered records, overview of File organizations, Indexing structures for files(Primary index, secondary index, clustering index), Multilevel indexing using B and B++ trees.

Books Recommended:

1.R. Elmasri, S.B. Navathe, Fundamentals of Database Systems 6th Edition, Pearson Education, 2010.

2.R. Ramakrishanan, J. Gehrke, Database Management Systems 3rd Edition, McGraw-Hill, 2002.

Database Management Systems Lab

1. Structured Query Language

Creating Database

Creating a Table

Specifying Relational Data Types

Specifying Constraints
Creating Indexes
2. Table and Record Handling
INSERT statement
Using SELECT and INSERT together
DELETE, UPDATE, TRUNCATE statements
DROP, ALTER statements
3. Retrieving Data from a Database
The SELECT statement
Using the WHERE clause
Using Logical Operators in the WHERE clause
Using IN, BETWEEN, LIKE , ORDER BY, GROUP BY and HAVING Clause
Using Aggregate Functions
Combining Tables Using JOINS
Subqueries
4. Database Management
Creating Views
Creating Column Aliases
Creating Database Users
Using GRANT and REVOKE

COSMI-4:
Introduction To Data Science With Python (40 Lectures)

Course Outcomes:
After completion of the course, learners should be able to
1. To understand the concept of data science and data science life cycle
2: To apply the pre-processing techniques for generating quality data inputs
3: To analyse the concept and parameters of exploratory data analytics
4: To develop the regression models using data science and analytics process
5: To analyse various tools and techniques of data visualization
6: handling data, encoding, tools apply, and types of data visualization.
.

Unit – I: Introduction

Introduction to Data Science – Evolution of Data Science – Role of Data Scientist, Data Science Life Cycle, Data Science Roles – Data Science Project Stages – Data Science Applications in Various Fields – Data Security Issues.

Unit – II: Data Collection and Data Pre-Processing

Data Collection Strategies – Need of Data Pre-processing,Pre-processing of data and data collection, Data Pre-Processing Overview – Data Cleaning – Data Integration and Transformation – Data Reduction – Data Discretization.

Unit – III: Exploratory Data Analytics

Introduction to Data Analytics/Concept of Data Analytics Types of Data Analytics, Descriptive Statistics – Mean, Standard Deviation, Skewness and Kurtosis – Box Plots – Pivot Table – Heat Map – Correlation Statistics – ANOVA.

Unit – IV: Model Development

Simple and Multiple Regression – Model Evaluation using Visualization – Residual Plot – Distribution Plot – Polynomial Regression and Pipelines – Measures for In-sample Evaluation – Prediction and Decision Making.

Unit – V: Model Evaluation

Generalization Error – Out-of-Sample Evaluation Metrics – Cross Validation – Overfitting – Under Fitting and Model Selection – Prediction by using Ridge Regression – Testing Multiple Parameters by using Grid Search.

REFERENCES:

1. Jojo Moolayil, “Smarter Decisions : The Intersection of IoT and Data Science”, PACKT, 2016.

2. Cathy O’Neil and Rachel Schutt , “Doing Data Science”, O'Reilly, 2015.

3. David Dietrich, Barry Heller, Beibei Yang, “Data Science and Big data Analytics”, EMC 2013

4. Raj, Pethuru, “Handbook of Research on Cloud Infrastructures for Big Data Analytics”, IGI Global.