

Academic Year: 2023-2024

Bachelor of Computer Application



**Syllabus & Scheme
Semester – I & II**

**School of Computer Science
& Applications**





GYANVEER UNIVERSITY, SAGAR (M.P.)
Scheme of Examination BCA II Semester (Major /Minor/Elective)
School of Computer Science (Academic Session 2023-24)
Subject wise distribution of marks and corresponding credits

S. No.	Course Type	Subject	Subject Code	Paper Name	Maximum Marks Allotted										Total Marks	Contact Periods Per week			Total Credits
					Theory Slot					Practical Slot						L	T	P	
					End Term Exam	Internal Assessment Class test (Descriptive & Objective)/Assignment/Seminar/			Internal Assessment			External Assessment							
						FINAL EXAM	Internal Assessment I	Internal Assessment II	Internal Assessment III	Class test/ Interaction	Attendance	Practical/ Presentation	Viva Voce	Lab Work					
1	Major	Computer Application	BCA121T	Data Structure	60	20	20	20	-	-	-	-	-	100	4	0	0	4	
2	Major	Computer Application	BCA121P	Data Structure	-	-	-	-	10	10	20	10	50	100	0	0	2	2	
3	Minor	Computer Application	BCA122T	Programing Methodology & Object Oriented Programing C++	60	20	20	20	-	-	-	-	-	100	4	0	0	4	
4	Minor	Computer Application	BCA122P	Programing Lab C++	-	-	-	-	10	10	20	10	50	100	0	0	2	2	
5	Elective	Computer Application	BCA123T	Computational Mathematics	60	20	20	20	-	-	-	-	-	100	4	0	0	4	
8	AECC	Foundation Course	AECC121T	Hindi Language	50	-	-	-	-	-	-	-	-	50	2	0	0	2	
			AECC122T	Yoga & meditation	50	-	-	-	-	-	-	-	-	-	50	2	0	0	2

Total of Credits is 6+6+4+4 = 20

Note*: Allotment of Marks for Internal Assessment for theory portion is Best of Two / either of two and addition of them.

*Student elect One Subject Elective Subject out of Elective subject.



BCA- Semester-II
BCA121T- Data Structures

Objectives

- Develop simple algorithms and flow charts to solve a problem with programming using top down design principles.
- Writing efficient and well-structured computer algorithms/programs
- Learn to formulate iterative solutions and array processing algorithms for problems
- Use recursive techniques, pointers and searching methods in programming
- Know the contributions of Indians in the field of programming and data structures.

Course Content:

Unit-I:

(Lecture – 12)

Data Structure: Basic concepts, Linear and Non-Linear data structures

Algorithm Specification: Introduction, Recursive algorithms, Data Abstraction, Performance analysis.

Arrays: Representation of single, two-dimensional arrays, triangular arrays, sparse matrices-array and linked representations.

Unit-II

(Lecture – 12)

Stacks: Operations, Array and Linked Implementations, Applications-Infix to Postfix Conversion, Infix to Prefix Conversion, Postfix Expression Evaluation, Recursion Implementation.

Queues: Definition, Operations, Array and Linked Implementations.

Circular Queue-Insertion and Deletion Operations, Dequeue (Double Ended Queue), Priority Queue- Implementation.

Unit-III

(Lecture – 12)

Linked Lists : Singly Linked Lists, Operations, Concatenating, circularly linked lists-Operations for Circularly linked lists, Doubly Linked Lists-Operations, Doubly Circular Linked List, Header Linked List

Trees: Representation of Trees, Binary tree, Properties of Binary Trees, Binary Tree Representations-Array and Linked Representations, Binary Tree Traversals, Threaded Binary Trees. **Heap:** Definition, Insertion, Deletion

Unit-IV

(Lecture – 12)

Graphs: Graph ADT, Graph Representations, Graph Traversals, Searching.

Hashing: Introduction, Hash tables, Hash functions, Over flow Handling.

Sorting: Bubble Sort, Selection Sort, Insertion Sort, Quick Sort, Merge Sort, Comparison of Sorting Methods,

Search Trees: Binary Search Trees, AVL Trees- Definition and Examples.

Unit-V

(Lecture – 12)

Indian Contribution to the field: Innovations in India, origin of Julia Programming Language, Indian Engineers who designed new programming languages, open source languages, Dr. Sartaj Sahni - computer scientist - pioneer of data structures, Other relevant contributors and contributions.



**Learning
Outcomes:**

- Upon successful completion of the course, student will learn storing pattern of the data and appropriate search, sorting procedure for their access when required.

TextBooks:

- J. R. Hanly and E. 13. Koffman, "Problem. Solving and Program Design in C", Pearson, 2015
- E. Balguruswamy, "C++ ", TMH Publication ISBN 0-07-462038-X
- SHerbert Schildt, 'C++ The Complete Reference "TINAH Publication ISBN 0-07 463880-7

**Reference
Books:**

- Robert L. Kruse, "Data Structures and Program Design in C++", Pearson.
- D.S. Malik, Data Structure using C++", Second edition, Cengage Learning.
- M. A. Weiss, "Data structures and Algorithm Analysis in C", 2nd edition,

**Additional
Electronic
Reference
Material:**

https://www.tutorialspoint.com/web_development_tutorials.htm



BCA- Semester-II

BCA122T - Programing Methodology & Object Oriented Programing C++

Objectives:

- Understanding the concept and underlying principles of Object-Oriented Programming.
- Understanding how object-oriented concepts are incorporated using C++ programming language.
- Developing problem-solving and programming skills using OOP concept.
- Understanding the benefits of a well structured program.
- Developing the ability to solve real-world problems through software development in high-level programming language like C++ with file handling concept.

Course Content:

Unit-I:	Introduction: Introducing Object-Oriented Approach, Relating to other paradigms (functional, data decomposition). Basic terms and ideas: Abstraction, Encapsulation, Inheritance, Polymorphism, Basic programming of C++.	(Lecture- 12)
Unit-II	Classes and Objects: Encapsulation, abstract data types, Object & classes, attributes, methods, C++ class declaration, State identity and behavior of an object, Constructors and destructors, object types, Metaclass, abstract classes.	(Lecture- 12)
Unit-III	Inheritance: Access specifiers, Types of inheritance, Ambiguity resolution in Multiple Inheritance, Constructor calling (Implicit and Explicit Constructor call) to base class, Containership and inheritance, Virtual Base Class.	(Lecture- 12)
Unit-IV	Friend: Friend Function, Friend Member Function and Friend Class. Polymorphism: Function Overloading, Operator overloading, operator overloading using Friend. Virtual function & Pure Virtual function.	(Lecture- 12)
Unit-V	File Handling: Stream Classes Hierarchy, Opening and closing FILE, Read and write in file. File pointers and Manipulations, Error Handling in File Operation, Command line Argument.	(Lecture- 12)
Learning Outcomes:	• Upon successful completion of the course, Student will be able to write computer programs using OOP features and specific functions available in C++	



**Reference
Books:**

- Parsons D., Object Oriented Programming with C++, BPB Publication.
- Steven C. Lawlor, The Art of Programming Computer Science with C++, Vikas Publication.
- Schildt Herbert, C++: The Complete Reference, Tata McGraw Hill.
- Tony Gaddis, Watters, Muganda, Object-Oriented Programming in C++, Dreamtech.
- Venugopal A.R. & Rajkumar, T. Ravishanker, Mastering C++, Tata McGraw Hill.
- Lippman S. B. & Lajoie J., C++ Primer, Addison Wesley.
- Lafore R., Object Oriented Programming using C++, Galgotia.

**Additional
Electronic
Reference
Material:**

https://www.w3schools.com/cpp/cpp_oop.asp

https://www3.ntu.edu.sg/home/ehchua/programming/cpp/cp3_OOP.html



BCA- Semester-II
BCA123T - Computational Mathematics

Objectives:

- Implement trigonometric solutions for measurements in real world scenarios
- Implement matrices and simultaneous equations to solve complex problems
- Use statistical tools efficiently
- Use Mathematical Logic and Predicate calculus for solving problems

Course Content:

Unit-I: (Lecture – 12)

Trigonometry: Angles & their Measurement, Values of Trigonometric Ratios, Height and Distances.

Elementary Matrices and types of matrices.

Unit-II (Lecture – 12)

Equations: Simultaneous linear equations, Methods of Solving Simultaneous equations, Quadratic equations.

Unit-III (Lecture – 12)

Statistics: Frequency Distribution, Measure of Central Tendency: Mean, Mode, Median. Measures of variation: Mean deviation Standard Deviations.

Unit-IV (Lecture – 12)

Mathematical Logic : Statements, and notations, Connectives: Negation, Conjunctions, And Disjunction. Statement formulas and truth tables. Tautologies. Tautological implications, contradiction contingency.

Unit-V Set Theory : Basic concepts of set theory, notation inclusion and equality of sets the power set, types of sets, operations on set, Venn diagrams.

Learning Outcomes:

- Upon successful completion of the course, the students will learn the computational mathematics and trigonometric solutions for measurements in real world scenarios, Mathematical Logic and Predicate calculus for solving problems

Text Books:

- Business Mathematics S.M. SHUKLA, Sahitya Bhawan Publications.
- Business Mathematics D C Agrawal, Sree Sai Prakashan.
- S.K. Sarkar : A Text Books of Discrete Mathematics, S Chand, 2005
- A text book of Discrete Mathematics, 9/E. Sarkar S.K., S. Chand New Delhi 2016
- मध्यप्रदेश हिन्दी ग्रंथ अकादमी से प्रकाशित विषय से संबंधित पुस्तकें।

Reference Books:

- Fundamental of Statistics ELHANCE & ELHANCE, Kitab Mahal Publication.
- Mathematical Statistics, 8/E, RAY and SHARMA, Ram Prasad Sons.
- Business Mathematics, J.K. Singh. Himalaya Publishing House 2017
- Business Mathematics, 9/E Sancheti and Kapoor, Sultan Chand & Sons 2014

Additional Electronic Reference Material:

- <https://nptel.ac.in/courses/>
- <https://freevideolectures.com/university/iit-roorkee/>
- <https://www.highereducation.mp.gov.in>
- <https://epathshala.ncert.org.in>



BCA- Semester-II
BCA121P - Data Structures LAB

Objectives

- Develop simple algorithms and flow charts to with Programming using top down design principles.
- Writing efficient and well-structured algorithms/programs.
- Learn to formulate iterative solutions and algorithms for problems,
- Use recursive techniques, pointers and searching Programming
- Possess ability to choose a data structure to suitably Model any data used in computer applications.

Course Content:

Given the problem statement, students are required to formulate problem, develop flowchart/algorithm. execute and test it Students should be given assignments on following :

1. Write a program to implement push and pop operations on a stack using
2. Array.
3. Write a program to perform insert and delete operations on a queue using
4. Array.
5. Write a program for Linear search.
6. Write a program Binary search.
7. Write a program Bubble sort.
8. Write a program Selection sort.
9. Write a program for Quick sort.
10. Write a program for Insertion sort.
11. Write a program to implement linked list.

Text Books:

- J. R. Hanly and E. 13. Koffman, "Problem. Solving and Program Design in C", Pearson, 2015
- E. Balguruswamy, "C", TMH Publication ISBN 0-07-462038-X Herbert Schildt, C++ The Complete Reference "TINAH Publication ISBN 0-07-463880-7

Reference Books:

- R. Lafore, 'Object Oriented Programming C4-1-'
- N. Dale and C. Weems, 'Programming and problem solving with C++ brief edition', Jones & Bartlett
- Learning.
- Adam Drozdek, "Data Structures and algorithm in C", Third Edition, Cengage Learning.
- Sartaj Sahani, "Data Structures, Algorithms and Applications with C", McGraw Hill.
- Robert L. Kruse, "Data Structures and Program Design in C", Pearson.
- D.S. Malik, Data Structure using C", Second edition, Cengage Learning.
- M. A. Weiss, "Data structures and Algorithm Analysis in C", 2nd edition, Pearson.
- Li Ischutz, "Schatun's outline series Data structures", Tata McGraw-Hill

Web Link:

<https://nptel.ac.in/courses>



BCA- Semester-II

BCA122P - Programing Lab C++

Objectives:

- Developing the ability to study the basic concepts of Object oriented Programming. Students will be able to write programs in C++.
- Becoming familiar with the fundamentals and acquire programming skills in the C++ language.
- Developing efficient program for operator and function overloading concept
- Developing efficient program for friend function and virtual function.
- Demonstrate the use of various OOPs concepts with the help of programs.

Course Content:

(Total Hours - 35)

1. Program illustrating basic input/output operations using CIN, COUT.
2. Implementing class and objects.
3. Implementing function overloading.
4. Implementing various constructors and destructor
5. Program illustrating overloading of various operators.
6. Program illustrating use of Friend, Inline, Static Member functions, default arguments.
7. Program illustrating various forms of Inheritance
8. Program illustrating use of virtual functions, virtual Base Class.
9. Program illustrating use of file handling
10. Write a Program to design a class having static member function named show count () which has the property of displaying the number of objects created of the class
11. Write a Program which creates & uses array of object of a class.(for eg. implementing the list of Managers of a Company having details such as Name, Age, etc..).
12. Write a Program to find Maximum out of Two Numbers using friend function. Note: Here one number is a member of one class and the other number is member of some other class.
13. Write a Program to swap private data members of classes named as class_1, class_2 using friend function.
14. Write a Program using copy constructor to copy data of an object to another object.
15. Write a Program to allocate memory dynamically for an object of a given class using class's constructor.
16. Write a Program to design a class to represent a matrix. The class should have the functionality to insert and retrieve the elements of the matrix.
17. Write a program to design a class representing complex numbers and having the functionality of performing addition & multiplication of two complex numbers using operator overloading.
18. Write a Program to overload operators like *, <> using friend function. The following overloaded operators should work for a class vector.
19. Write a program for developing a matrix class which can handle integer matrices of different dimensions. Also overload the operator for addition, multiplication & comparison of matrices.
20. Write a program to overload new/delete operators in a class.



21. Write a program in C++ to highlight the difference between overloaded assignment operator and copy constructor.
22. Write a Program illustrating how the constructors are implemented and the order in which they are called when the classes are inherited. Use three classes named alpha, beta, gamma such that alpha, beta are base class and gamma is derived class inheriting alpha & beta.
23. Write a program to maintain the records of person with details (Name and Age) and find the eldest among them. The program must use this pointer to return the result.
24. Write a Program to illustrate the use of pointers to objects which are related by inheritance.
25. Write a program illustrating the use of virtual functions in class.
26. Write a program to show conversion from string to int and vice-versa.
27. Write a program showing data conversion between objects of different classes.
28. Write a program showing data conversion between objects of different classes and conversion routine should reside in destination class.
29. Write a program to copy the contents of one file to another.
30. Write a program to perform read/write binary I/O operation on a file (i.e. write the object of a structure/class to file).
31. Write a program to maintain a elementary database of employees using files.
32. Write a program for reading and writing data to and from the file using command line arguments.
33. Write a program showing implementation of stack class having the functionality of push, pop operations.
34. Write program to implement a queue class with required operations/ functions.
35. Write a program to implement circular queue class with required operations/ functions.

Learning Outcomes:

- Upon successful completion of the course, students will be competent enough to write program in C++ on the problems assigned.

Reference Books:

1. Schildt Herbert, C++: The Complete Reference, Tata McGraw Hill.
2. Tony Gaddis, Watters, Muganda, Object-Oriented Programming in C++, Dreamtech.
3. Venugopal A.R. & Rajkumar, T. Ravishanker, Mastering C++, Tata McGraw Hill.

*** Latest editions of all the suggested books are recommended.**

Additional Electronic Reference Material:

https://www.w3schools.com/cpp/cpp_oop.asp

https://www3.ntu.edu.sg/home/ehchua/programming/cpp/cp3_OOP.html

