

Morigaon College (Autonomous)

Morigaon : Assam



SYLLABUS OF Bachelor of Computer Applications (BCA) (Upto 3rd Semester)

Approved By:

Sl. No.	Name	Designation	Signature
1	Ms. Nabanika Sarkar	Chairperson	Nabanika Sarkar
2	Prof. Anjana Kakoti Mahanta	GU Nominee	
3	Dr. Bornali Gogoi	Subject Expert	Bornali Gogoi 19/12/2025
4	Dr. Mousoomi Bora	Subject Expert	Mousoomi Bora
5	Mr. Pranab Nath	Industry Expert	
6	Mr. Ripunjoy Deka	Alumni	Ripunjoy Deka
7	Ms. Harshita Hazarika	Member	Harshita Hazarika
8.	Ms. Nisha Patar	Member	Nisha Patar

SYLLABUS OF

Bachelor of Computer Applications

Morigaon College (Autonomous)

<u>NEP Syllabus</u>			
BCA 1st Semester			
Sl. No.	Paper Name	Course Type	Credit
1	Computer Fundamentals and C- Programming	Compulsory	4(3+1)
2	Mathematics I	Compulsory	4
3	Web Technologies (SEC-1)	SEC	3 (2+1)
4	AEC-1 (Alt. English)	AEC	4
5	MDC-1	MDC	3
6	VAC-1	VAC	2
BCA 2nd Semester			
1	Digital Logic Fundamentals	Compulsory	4
2	Data Structure & Algorithms Using C	Compulsory	4(3+1)
3	Business Accountancy & Intelligence	SEC	3(2+1)
4	AEC-2 (Communicative English)	AEC	4
5	MDC-2	MDC	3
6	VAC-2	VAC	2
BCA 3rd Semester			
1	Computer organization and Architecture	Compulsory	4
2	Mathematics II	Compulsory	4
3	Object Oriented Programming in C++	Compulsory	4(3+1)
4	Information and Cyber Security	SEC	3
5	MDC-3	MDC	3
6	VAC-3	VAC	2

BCA 1st Semester

Computer Fundamentals and C-Programming(Core/Credit-4)

Course Code: BCA-CORE-MC-0100104

1. Learning Outcomes: At the end of the course, students will be able to:
 - (a) Understand the fundamentals of Computer and its components
 - (b) Understand the basics of C programming like data types and operators
 - (c) Understand and write program in C to implement conditions, loops, functions
 - (d) Work on arrays, strings and basic file operations
2. Total Credit: 4, Theory: 3, Practical: 1
3. Reference books:
 - (a) Fundamentals of Computers, E Balagurusamy, McGraw Hill Education
 - (b) Computer Fundamentals, Anita Goel, Pearson Education
 - (c) B.S. Gottfried, "Schaum's Outline of Theory and Problems of Programming with C", Mcgraw-Hill, 2007.
 - (d) B. Kernighan, D. Ritchie, "The C Programming Language", Second Edition, Prentice Hall, 1988
 - (e) E. Balaguruswami, "Programming in ANSI C", 2nd Ed., Tata McGraw Hill, 2004.
4. Detailed Syllabus:

A. Theory

Unit 1: Introduction to Computers

Definition of computer, basic components of computer, bus, evolution of computers, Generations of computers, data representation in a computer, ASCII, Unicode, Number system: decimal, binary, octal and hexadecimal number system, conversion among number systems, Memory: primary and secondary memory, five levels of memory hierarchy.

Unit 2: Getting started with C programming

Introduction to programming languages: High-level vs low level languages, compiled vs interpreted languages, definition of pseudocode, flowchart, flowchart of algorithm with examples.

Structure of a C program. Introduction to Header files. Main function and a simple program execution. Compiling and executing a program. C tokens – keywords, identifiers, constants, operators. Statements and expressions in C. Basic data types in C - integers, floats, doubles, characters, void. Size and range of values of data types. Variables, Constants – integer constant, real constant, character constant, string constant. Declaration and initialization of variables and constants. Assigning values to variables. Operators in C – binary and unary operators. Arithmetic, assignment, logical, comparison, bitwise and conditional operators. Order of precedence of operators. Associativity of operators. Input and output statements – getchar(), getc(), getch(), putchar(), putc(), puts(), scanf(), printf(), format specifiers. Typecasting.

Unit 3: Control Structures in C

Control Structures in C. Basic programming constructs- Sequence, selection and iteration. Conditional statements – if, else, switch case. Nested conditions. Loops – for loop, while loop, do- while loop. Using loop for counting iterations. Using while loop for indefinite iterations. Nested loops. Break and continue

statements.

Unit 4: Arrays and Strings

Introduction to Arrays. Declaration and initialization of arrays. Accessing array elements. Multidimensional arrays. Introduction to Strings. Declaration and initialization of strings. String input and output in C.

Unit 5: Functions and Pointers

Introduction to functions. Function declaration and definition. Return types of function. Function arguments. Function calling – call by value vs call by reference. Passing an array as argument to a function. Basic concept of recursion.

Introduction to Pointers. Pointer declaration and initialization. Pointers and addresses. Pointers and Arrays. Basic concept of dynamic memory allocation, malloc(), calloc().

Unit 6: Introduction to Structures and Unions

Basic concept of Structures and Unions in C. Structure declaration and initialization. Union declaration and initialization. Difference between structures and unions.

Unit 7: File Processing in C

Basic concept of file handling. Opening and closing file using fopen() and fclose(). Binary vs text files. Reading and writing files – fgets(), fscanf(), fprintf(). Random access to files.

B. List of Practical

1. Write a C program to display Hello World.
2. Write a C program to add two numbers entered by the user.
3. Write a C program to check whether a number is even or odd.
4. Write a C program to find the largest of three numbers.
5. Write a C program to check whether a given year is a leap year.
6. Write a C program to generate the multiplication table of a number.
7. Write a C program to find the factorial of a number using loop.
8. Write a C program to generate Fibonacci series.
9. Write a C program to check whether a number is positive, negative, or zero.
10. Write a C program to demonstrate switch–case for simple calculator.
11. Write a C program to find the largest element in an array.
12. Write a C program to sort array elements in ascending order.
13. Write a C program to perform matrix addition.
14. Write a C program to check whether a number is prime.
15. Write a C program to reverse a number.
16. Write a C program to check whether a string is palindrome.
17. Write a C program to find length of a string without using library function.
18. Write a C program to swap two numbers using functions.
19. Write a C program to find factorial using recursion.
20. Write a C program to demonstrate array of structures to store student details.
21. Write a C program to access array elements using pointers.
22. Write a C program to copy contents of one file to another.
23. Write a C program to store and display student records using file handling

Mathematics I

(Core/Credit-4)

Course Code: BCA-CORE-MC-0100204

1. Learning Outcomes: After successful completion of this course, students will be able to:
 - (a) Learn the concepts of set, relation, and function from Computer Science point of view.
 - (b) Know how to view a table/database as an n-ary relation.
 - (c) Learn what a matrix is and relate it with arrays used in programming.
 - (d) Understand determinants and how determinants are used in solving simultaneous equations.
 - (e) Get familiar with statistical and probabilistic measures that are used in computation related software/packages.
2. Total Credit: 4, Theory: 4, Practical: 0
3. Reference books:
 - (a) Discrete Mathematics Structures with Applications to Computer Science, J. P. Tremblay and R. Manohar, Mc-Graw Hill.
 - (b) Discrete Mathematics, N. Ch.SN Iyengar, K.A. Venkatesh, V. M. Chandrasekaran, P. S. Arunachalam, Vikash Publishing House Pvt Ltd.
 - (c) Elements of Discrete Mathematics, C. L. Liu, Mc-Graw Hill International Ed.

4. Course Details:

Unit 1: Sets, Relations and Functions

Sets: definition of set, cardinality of sets, finite, countable and infinite sets. Operations on sets, Venn diagram. Principle of inclusion and exclusion and their applications on simple problems. Multisets.

Relations: Definition and properties of binary relations, closures of relations, equivalence relations, equivalence classes and partitions, n-ary relations and representation of n-ary relations as tables. Partial ordering relations and lattices,

Functions: Definition of function, one-to-one and onto, principles of mathematical induction. Concave and convex functions.

Unit 2: Matrices

Definition and different types (such as identity matrix, diagonal matrix etc) of matrices, row and column operations; vectors and matrices, Addition, subtraction and multiplication of matrices, Properties of matrix operations, Existence of additive and multiplicative identity and additive inverse of a matrix. Representing relations using matrices. Transpose of a matrix and its properties. Symmetric and skew symmetric matrices, Elementary transformation of a matrix, Invertible matrices.

Unit 3: Determinants

Determinant of a square matrix, minor, cofactor, Adjoint of a matrix and matrix inversion. Inverse of a matrix using elementary transformation. Rank of a matrix and determination of rank of a matrix. Eigen values and Eigen vectors of a matrix (Stressing on symmetric matrices), Cayley-Hamilton theorem – Cramer's rule, Consistency of a system of linear non-homogenous equations and existence of solutions (statement only), Simple problems, Solutions of simultaneous linear equations by Gaussian elimination method.

Unit 4: Fundamentals of Statistics and Discrete Probability

Types of Data, Attributes and variables; Construction of Frequency, Cumulative frequency. Graphical representation of Frequency distribution: Histogram, Frequency Polygon, Frequency Curve and Cumulative Frequency curves (Ogive curves). Diagrammatic representations: Simple bar, Subdivided bar, Pie diagrams.

Measures of central tendency-Mean, Median and Mode. Measures of variation – Range, Interquartile range, Standard Deviation and Variance.

Sample space, events, random variables, basic probability. Conditional Probability and Bayes theorem

Web Technologies (SEC/Credit-3)

Course Code: SEC-BCA-MC-0100103

1. Learning Outcome: At the end of the course, students will be able to:
 - Understand the basic concept of web applications and web services.
 - Design basic well-structured web page using HTML and CSS
 - Develop the ability to implement interactive elements and dynamic content using basic JavaScript

2. Total Credit: 3, Theory: 2, Practical: 1

3. List of Reference Books:

- a) Jackson J.C. (2007). *Web Technologies: A Computer Science Perspective*. Pearson.
- b) Duckett, J. (2011). *HTML and CSS: Design and Build Websites*. John Wiley & Sons.
- c) Robbins, J. N. (2018). *Learning Web Design: A Beginner's Guide*. O'Reilly Media.
- d) Haverbeke, M. (2018). *Eloquent JavaScript*. No Starch Press.
- e) Welling, L., & Thomson, L. (2016). *PHP and MySQL Web Development* (5th ed.). Addison-Wesley Professional.

4. Detailed Syllabus:

A. Theory

Unit 1: Introduction to Web Technologies

Concepts of Internet and WWW, Web browsers and their functionalities, Client–Server architecture, HTTP vs HTTPS (basic), Brief introduction to FTP, Ports, URL, cache, cookies, Basic working of DNS, Types of websites – static vs dynamic, Basic idea of domain & hosting

Unit 2: Front End Development using HTML

HTML structure and basic tags, Text formatting tags, Links and Navigation, Lists – ordered, unordered, definition, Images and multimedia (basic), Tables (simple), Forms – text, email, password, radio, checkbox, select, textarea, button, Semantic HTML tags – header, nav, section, article, footer

Unit 3: Front End Design using CSS

Introduction to CSS, Inline, Internal, External CSS, Basic Selectors – element, class, ID, Box Model – content, padding, border, margin, Colors & background basics, Typography – fonts and text properties

Unit 4: Client-Side Scripting with JavaScript

Introduction to JavaScript, JavaScript syntax and datatypes, Variables – var, let, const, Operators, Conditional statements, Loops, Functions – declaration and usage, DOM Basics – getElementById, querySelector, Changing content and basic style manipulation, Simple onclick events

B. List of Practical

1. Create a basic HTML webpage structure with a heading, paragraph, and an image.
2. Build a navigation menu using an unordered list () with clickable links.
3. Implement a form with input fields for name, email, and a submit button.
4. Create a table with multiple rows and columns to display tabular data.
5. Design an image gallery using HTML and CSS with proper padding and border.
6. Embed a YouTube video on a webpage using the <iframe> tag.
7. Implement an ordered list () to display a step-by-step tutorial or instructions.
8. Create a dropdown select menu (<select>) with multiple options.
9. Build a registration form with fields for name, email, password, date of birth, address and other such fields with a submit button. Include appropriate input types, labels and placeholders.

10. Style a heading element with a custom font, colour and background.
11. Implement a hover effect on a button that changes its background colour or adds a solid border.
12. Implement a CSS tooltip that displays additional information when hovering over an element..
13. Create a button that, when clicked, appends a new paragraph element with a specific text content to an existing div element..
14. Build a form with input fields for name and email.
15. Build a form with input fields for email, password and confirm password. When the form is submitted, use an alert to display a success message if the password and confirm password values matches, otherwise show an error alert. Use JavaScript for the validation.