# BACHELORS WITH COMPUTER APPLICATIONS AS MAJOR
## 2nd SEMESTER

## CAP222J: COMPUTER APPLICATIONS _ PROGRAMMING FUNDAMENTALS THROUGH C

**CREDITS: THEORY: 4; PRACTICAL: 2**

***COURSE OBJECTIVES:***

1. *To demonstrate the use of flowcharts and algorithms for problem solving*
2. *To introduce the concepts of structured programming*
3. *To familiarize the student with the syntactic constructs of C*
4. *To enable to the students to translate algorithms into C programs*

### THEORY (4 CREDITS)

**UNIT – I** **(15 Lectures)**

**Programming Languages:** History and Role of Programming Languages, Syntax and semantics, source code and object code, datatypes, variables, constants, declaration, Structured Data Types. Sequence Control: Implicit and Explicit. Sequence control between Statements. Subprogram Control: Simple call return and recursive subprogram. Language Paradigms: Simple Procedural Languages, Block Structured Programming Languages, Object Based Languages, Functional Languages, Logic Programming Languages.

Flowcharts, Flowchart Elements, Problem Solving Through Flowcharts.

Algorithms, Characteristics of an Algorithm, Algorithms for basic problems.

**UNIT – II**

**Introduction to C Programming** **(15 Lectures)**

History and overview of C, Basic structure of a C Program, Compilation, Execution and Debugging of programs in C.

Keywords, Identifiers and Datatypes. Variables and Constants. Comments. Console I/O using printf() and scanf(). Typecasting.

Operators – Arithmetic, Logical, Relational, Increment Decrement and Assignment Operators. Expressions. Operator Precedence.

Conditional Statements (If, If-Else, If-Else If, Nested If, Switch).

**UNIT – III** **(15 Lectures)**

**Looping, Functions and Pointers.**

Loops (while, do-while, for). Break and Continue. Nested Loops.

Functions: Declaring, Defining and Calling. Call by Value, Call by Reference. Function Arguments and Return Values.

Pointers: Declaring and Initializing. Accessing value of a pointer variable. Pointer Expressions. Pointer Increments and Scale Factors. Pointers and Arrays. Passing Pointers to Functions.

**UNIT – IV**

**Arrays, Strings, Structures and Unions**

Declaring, Initializing 1-D arrays and 2-D arrays. Accessing Elements of an Array, Memory Layout of Arrays. Passing Arrays to Functions, Command Line Arguments.

Character Arrays and String. Declaring and Initializing Strings, Reading and Writing Strings, String Handling Function (strlen, strcat, strcmp, strcpy).

Structures and Unions: Declaring, initializing and using simple structures and unions; Manipulating individual members of structures and unions, Array of structures, Passing structures to functions.

Dynamic Memory Allocation using malloc and free.

### TEXTBOOKS:

1. Terence Pratt, Programming Languages Design and Implementation (Pearson/Prentice Hall)
2. Balagurusamy, Programming in ANSI C, 8th Edition (McGraw Hill)

### REFERENCES:

1. Kanetkar - Let us C (BPB Publications)
2. Ghezzi, Jazayeri - Programming Language Concepts (Wiley)
3. Srivastava – C in Depth (BPB Publications)
4. Beej's Guide to C Programming
5. Byron Gottfried - Schaum's Outline of Programming with C (McGraw-Hill)

**LABORATORY COURSE (02 CREDITS):**

1. WAP to print the sum of digits of an integer.
2. WAP to print the product of digits of an integer.
3. WAP to reverse a number.
4. WAP to compute the sum of the first n terms of the following series S = 1+1/2+1/3+1/4+……
5. WAP to compute the sum of the first n terms of the following series S =1-2+3-4+5…………….
6. Write a program to check whether a given string is Palindrome or not. Convert this program into to a function that checks if a given string is a palindrome.
7. Write a function to find whether a given no. is prime or not. Use the same to generate the prime numbers less than 100.
8. WAP to compute the factors of a given number.
9. Write a macro that swaps two numbers. WAP to use it.
10. WAP to print a triangle of stars as follows (take number of lines from user):

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

11. 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 the array

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.

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 create an array of user-defined size dynamically using malloc() function. Display the sum of values entered in it and use free() to release its memory.
16. Write a 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) Reverse the string

***

## BACHELORS WITH APPLIED COMPUTING AS MINOR

*(FOR STUDENTS WITH MAJOR IN COMPUTER APPLICATIONS / INFORMATION TECHNOLOGY)*

**2nd SEMESTER**

## ACP222N: APPLIED COMPUTING _ WEB DESIGNING

**CREDITS: 4 + 2**

### UNIT-I

Markup Languages, Introduction to HTML5, Development Environment Setup, Anatomy of an HTML Tag, Basic Structure of HTML Document, HTML Content Models, Meta-Tags, Formatting Tags, Text Level Formatting, Lists, Hyperlinks, Image and Image Maps, Table Tags, HTML Comment tag. Block and inline elements, redirecting to another URL, creating division-based layouts. Forms: creating basic form, using check boxes, textboxes and option buttons, input validation and additional input types in HTML5, HTML multimedia basics. HTML DOM structure.

### UNIT II:

Need for CSS. Different approaches to style sheets, Anatomy of a CSS Rule. Element, Class, and ID Selectors. Combining Selectors, Pseudo-Class Selectors. Style Placement, Conflict Resolution, Styling Text Wildcard Selectors (*, ^ and $) in CSS. Web fonts. Working with Browser Developer Tools.CSS Box Model: - background, margin, padding, Float and z-index properties, Relative and Absolute Element Positioning. Basic Introduction to Bootstrap Framework.

### UNIT-III

Introduction to Javascript, Different approaches to place Javascript code in an HTML File. JS identifiers, Reserved Words, Optional Semicolons, Comments, Literals. Types, Values and Variables: Numbers, Text, Booleans. Nulls and undefined. Type Conversions. Variable Declaration and Assignment. Const, let and var.
Expressions and Operators: Arithmetic, Relational, Logical, Assignment and Evaluation Expressions. Conditionals: if, else if and switch. Loops: while and for. Break, continue, return and yield.
Functions: Defining, Invoking, Function Arguments and Parameters. Functions as Values.

### UNIT-IV

Objects: Creating Objects, Querying and setting Properties, Deleting and Testing Properties. Serializing Objects. Arrays: Creating, Reading, Writing arrays. Array length. Iterating Arrays, Strings as Arrays.
The Document Object Model, Program Input and Output, Browser Events and Event Handling.

### RECOMMENDED BOOKS:

1. Jennifer Robbins - Learning Web Design: A Beginner’s Guide to HTML, CSS, JavaScript, and Web Graphics (5e, 2018, O’Reilly Media)
2. Terry Felke-Morris - Web Development & Design Foundations with HTML5 (8e, 2017, Pearson)
3. Eric Meyer, Estelle Weyl - CSS The Definitive Guide (4e, 2018, O'Reilly Media)
4. David Sawyer McFarland - CSS The Missing Manual (4e, 2015, O'Reilly Media)
5. David Flanagan - JavaScript_ The Definitive Guide (7e, 2020, O'Reilly Media)
6. Cay S. Horstmann - Modern JavaScript for the Impatient (Addison-Wesley Professional, 2020)

### WEB DESIGNING LAB

1. Design a Home page for your college
2. Design a web page with links to different pages and allow navigation between web pages.
3. Design a web page using Images
4. Use a HTML table to design a page with a header, sidebar, main content and footer.
5. Design a user registration form using different HTML form controls
6. Design a web page with buttons that can handle different page events using JS event handlers.
7. Use Java Script to change the image displayed in an img tag when a button on the page is clicked.
8. Use bootstrap to add formatting to your home page.
9. Write a JavaScript program with proper GUI to perform unit conversion using the onChange event.
10. Design the interface of a login page using HTML and CSS.
11. Design a simple “To Do” Application using HTML/CSS/JavaScript.
12. Design Basic Calculator using HTML/CSS/JavaScript.
13. Design and develop a simple “Tic-Tac-Toe Game” using HTML/CSS/JavaScript.
14. Remove a specific table row using Java Script.
15. Set value in input text using Java Script.
16. Set a value in a span using Java Script.

# BACHELORS WITH COMPUTER APPLICATIONS AS MINOR

*(FOR STUDENTS WITH SUBJECTS OTHER THAN COMPUTER APPLICATION / INFORMATION TECHNOLOGY AS MAJOR)*

## 2nd SEMESTER

## CAP222N: COMPUTER APPLICATIONS _ PROGRAMMING FUNDAMENTALS THROUGH C

**CREDITS: THEORY: 4; PRACTICAL: 2**

***COURSE OBJECTIVES:***

1. *To demonstrate the use of flowcharts and algorithms for problem solving*
2. *To introduce the concepts of structured programming*
3. *To familiarize the student with the syntactic constructs of C*
4. *To enable to the students to translate algorithms into C programs*

### THEORY (4 CREDITS)

**UNIT – I** **(15 Lectures)**

**Programming Languages:** History and Role of Programming Languages, Syntax and semantics, source code and object code, datatypes, variables, constants, declaration, Structured Data Types. Sequence Control: Implicit and Explicit. Sequence control between Statements. Subprogram Control: Simple call return and recursive subprogram. Language Paradigms: Simple Procedural Languages, Block Structured Programming Languages, Object Based Languages, Functional Languages, Logic Programming Languages.

Flowcharts, Flowchart Elements, Problem Solving Through Flowcharts.

Algorithms, Characteristics of an Algorithm, Algorithms for basic problems.

**UNIT – II**

**Introduction to C Programming** **(15 Lectures)**

History and overview of C, Basic structure of a C Program, Compilation, Execution and Debugging of programs in C.

Keywords, Identifiers and Datatypes. Variables and Constants. Comments. Console I/O using printf() and scanf(). Typecasting.

Operators – Arithmetic, Logical, Relational, Increment Decrement and Assignment Operators. Expressions. Operator Precedence.

Conditional Statements (If, If-Else, If-Else If, Nested If, Switch).

**UNIT – III** **(15 Lectures)**

**Looping, Functions and Pointers.**

Loops (while, do-while, for). Break and Continue. Nested Loops.

Functions: Declaring, Defining and Calling. Call by Value, Call by Reference. Function Arguments and Return Values.

Pointers: Declaring and Initializing. Accessing value of a pointer variable. Pointer Expressions. Pointer Increments and Scale Factors. Pointers and Arrays. Passing Pointers to Functions.

**UNIT – IV**

**Arrays, Strings, Structures and Unions**

Declaring, Initializing 1-D arrays and 2-D arrays. Accessing Elements of an Array, Memory Layout of Arrays. Passing Arrays to Functions, Command Line Arguments.

Character Arrays and String. Declaring and Initializing Strings, Reading and Writing Strings, String Handling Function (strlen, strcat, strcmp, strcpy).

Structures and Unions: Declaring, initializing and using simple structures and unions; Manipulating individual members of structures and unions, Array of structures, Passing structures to functions.

Dynamic Memory Allocation using malloc and free.

### TEXTBOOKS:

1. Terence Pratt, Programming Languages Design and Implementation (Pearson/Prentice Hall)
2. Balagurusamy, Programming in ANSI C, 8th Edition (McGraw Hill)

### REFERENCES:

1. Kanetkar - Let us C (BPB Publications)
2. Ghezzi, Jazayeri - Programming Language Concepts (Wiley)
3. Srivastava – C in Depth (BPB Publications)
4. Beej's Guide to C Programming

5. Byron Gottfried - Schaum's Outline of Programming with C (McGraw-Hill)

## LABORATORY COURSE (02 CREDITS):

1. WAP to print the sum of digits of an integer.
2. WAP to print the product of digits of an integer.
3. WAP to reverse a number.
4. WAP to compute the sum of the first n terms of the following series S = 1+1/2+1/3+1/4+……
5. WAP to compute the sum of the first n terms of the following series S =1-2+3-4+5…………….
6. Write a program to check whether a given string is Palindrome or not. Convert this program into to a function that checks if a given string is a palindrome.
7. Write a function to find whether a given no. is prime or not. Use the same to generate the prime numbers less than 100.
8. WAP to compute the factors of a given number.
9. Write a macro that swaps two numbers. WAP to use it.
10. WAP to print a triangle of stars as follows (take number of lines from user):

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

11. 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 the array

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.

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 create an array of user-defined size dynamically using malloc() function. Display the sum of values entered in it and use free() to release its memory.
16. Write a 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) Reverse the string

***

# 2nd SEMESTER
# SKILL ENHANCEMENT COURSE (SEC)

## CAP222S: PHP Programming (2+2 CREDITS)

**Introduction to PHP: (3L)**

PHP introduction, inventions and versions, important tools and software requirements (like Web Server, Database, Editors etc.)

PHP with other technologies, scope of PHP

Basic Syntax, PHP variables and constants

Types of data in PHP, Expressions, scopes of a variable (local, global)

PHP Operators: Arithmetic, Assignment, Relational, Logical operators, Bitwise, ternary and MOD operator.

PHP operator Precedence and associativity

**Handling HTML form with PHP: (2L)**

Capturing Form Data GET and POST form methods

Dealing with multi value fields

Redirecting a form after submission

**PHP conditional events and Loops: (3L)**

PHP IF Else conditional statements (Nested IF and Else)

Switch case, while, For and Do While Loop

Goto, Break, Continue and exit

**PHP Functions: (3L)**

Function, Need of Function, declaration and calling of a function

PHP Function with arguments, Default Arguments in Function

Function argument with call by value, call by reference

Scope of Function Global and Local

**String Manipulation and Regular Expression: (2L)**

Creating and accessing String , Searching & Replacing String

Formatting joining and splitting String , String Related Library functions

Use and advantage of regular expression over inbuilt function

Use of preg_match(), preg_replace(), preg_split() functions in regular expression

**Array: (2L)**

Anatomy of an Array ,Creating index based and Associative array ,Accessing array

Looping with Index based array, with associative array using each() and foreach()

**Some useful Library function Reference Books:**

1. Steven Holzner, "PHP: The Complete Reference Paperback", McGraw Hill Education (India), 2007.
2. Timothy Boronczyk, Martin E. Psinas, "PHP and MYSQL (Create-Modify-Reuse)", Wiley India Private Limited, 2008.
3. Robin Nixon, "Learning PHP, MySQL, JavaScript, CSS & HTML5", 3rd Edition Paperback, O'reilly, 2014.
4. Luke Welling, Laura Thompson, PHP and MySQL Web Development", 4th Edition, Addition Paperback, Addison-Wesley Professsional,2008.
5. David Sklar, Adam Trachtenberg, "PHP Cookbook: Solutions & Examples for PHP Programmers", 2014.

# Lab BCA-16404SE: PHP Programming

**Practicals: 30 Lectures**

1. Create a PHP page using functions for comparing three integers and print the largest number.
2. Write a function to calculate the factorial of a number (non-negative integer). The function accept the number as an argument.
3. WAP to check whether the given number is prime or not.
4. Create a PHP page which accepts string from user. After submission that page displays the reverse of provided string.
5. Write a PHP function that checks if a string is all lower case.
6. Write a PHP script that checks whether a passed string is palindrome or not? (A palindrome is word, phrase, or sequence that reads the same backward as forward, e.g., madam or nurses run)
7. WAP to sort an array.
8. Write a PHP script that removes the whitespaces from a string.
   Sample string : 'The quick "" brown fox' Expected
   Output:
   Thequick""brownfox
9. Write a PHP script that finds out the sum of first n odd numbers.
10. Create a login page having user name and password. On clicking submit, a welcome message should be displayed if the user is already registered (i.e.name is present in the database) otherwise error message should be displayed.
11. Write a PHP script that checks if a string contains another string.
12. Create a simple 'birthday countdown' script, the script will count the number of days between current day and birth day.
13. Create a script to construct the following pattern, using nested for loop.

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

14. Write a simple PHP program to check that emails are valid.
15. WAP to print first n even numbers.
16. $color = array('white', 'green', 'red")
    Write a PHP script which will display the colors in the following way:

    :Output:

    white, green, red,

    .

    green • red • white

17. Using switch case and dropdown list display a —Helloll message depending on the language selected in drop down list.
18. Write a PHP program to print Fibonacci series using recursion.
19. Write a PHP script to replace the first 'the' of the following string with 'That'.
    **Sample:** 'the quick brown fox jumps over the lazy dog.'
    **Expected Result:** That quick brown fox jumps over the lazy dog.

# 2$^{nd}$ SEMESTER
# COMPUTER APPLICATIONS
### (WEB DEVELOPER)
## SKILL ENHANCEMENT COURSE (SEC)

## WDP222S: JAVA SCRIPT AND CSS BASICS

**CREDITS: THEORY: 2, PRACTICAL: 2**

## THEORY (2 CREDITS)

### UNIT 1 – JAVA SCRIPT (15 LECTURES)

Introduction, Script Tag, Data Types, Variables, Literals, Expressions, Operators, Conditional Statements (if, if-else, if-else-if-else), switch-case, Looping Statements (while, for, do-while), Array, Associative Arrays, Functions, Event Handling, Javascript Objects (Browser, Document, Window etc.)

### UNIT 2 – CSS (15 LECTURES)

DHTML introduction, Style Sheets-Embedded Styles, Inline Styles, External Style Sheets, Using Classes, Style Sheet Properties- Fonts Properties, Background and Colour Properties, Text Properties, Box Properties, Classification Properties-Display Property, Whitespace Property, , CSS Units, URL's , DIV and SPAN Tags, Dynamic Positioning, Layering, DHTML Events.

REFERENCE BOOKS:

1. Java  Script Bible Wrox Publications
2. DHTML BPB Publications

## PRACTICAL (2 CREDITS)

### LAB SHEET- JAVA SCRIPT AND CSS BASICS

1. Write a JavaScript program to display the current day and time in the following format
2. Write a JavaScript program to get the current date
3. Write a JavaScript program that accept two integers and display the larger
4. Write a JavaScript conditional statement to sort three numbers. Display an alert box to show the result
5. Write JavaScript to demonstrate loops: while, for, do-while
6. Write a JavaScript for loop that will iterate from 0 to 15. For each iteration, it will check if the current number is odd or even, and display a message to the screen
7. Write a JavaScript function to check whether an `input` is an array or not
8. Write a JavaScript program to sort the items of an array
9. WAP to show blinking effect on a web page using JavaScript.
10. Write CSS for *Fonts, Background, Color, Text*
11. Design a digital clock using JavaScript and CSS.
12. Design a calculator using HTML & JavaScript.
13. Write a JavaScript program to demonstrate Event Handling.
14. WAP to validate Email Address in JavaScript.
15. Write a program to demonstrate exception handling in JS.