

## CPS-110: COMPUTER FUNDAMENTALS AND C PROGRAMMING LAB

|                                  |                                           |                    |           |
|----------------------------------|-------------------------------------------|--------------------|-----------|
| Course Title                     | Computer Fundamentals & C Programming Lab | Course Code        | CPS-110   |
| YEAR                             | 1 <sup>st</sup> YEAR                      | CourseGroup        | Practical |
| TeachingScheme in Periods(L:T:P) | 0:0:3                                     | Credits            | 2         |
| Methodology                      | Lecture+Practical                         | TotalContactHours: | 90        |
| CIE                              | 60Marks                                   | SEE                | 40Marks   |

### Prerequisites

Basic knowledge of Mathematics, Logical, Reasoning and IT skills.

### CourseOutcomes

**On successful completion of the course, the students will be able to**

| CO'S       | Description                                                                                                              |
|------------|--------------------------------------------------------------------------------------------------------------------------|
| <b>CO1</b> | <b>Identify</b> computer hardware components and <b>perform</b> OS installation and file management tasks.               |
| <b>CO2</b> | <b>Create</b> professional documents, analyze data using spreadsheets, and <b>design</b> effective presentations.        |
| <b>CO3</b> | <b>Apply</b> C syntax and operators to solve basic arithmetic and decision-making problems.                              |
| <b>CO4</b> | <b>Develop</b> modular algorithms using loops and functions for mathematical logic                                       |
| <b>CO5</b> | <b>Construct</b> programs using Arrays, Strings, and Bitwise operators for complex data and hardware-level manipulation. |

**UNIT Plan:**

| Unit No. | UnitName                                      | Periods |
|----------|-----------------------------------------------|---------|
| 1        | Computer Hardware and Operating System Basics | 12      |
| 2        | Office Automation Tools                       | 18      |
| 3        | Introduction to C Programming                 | 15      |
| 4        | Control Structures and Functions              | 20      |
| 5        | Advanced C Programming                        | 25      |
|          | Total                                         | 90      |

**Recommended Books**

- |                            |    |                                     |
|----------------------------|----|-------------------------------------|
| 1.LetUsC                   | -- | Yeshwanth Kanetkar BPB Publications |
| 2.Programming in ANSI C    | -- | E.Balaguruswamy Tata McGraw Hill    |
| 3.Programming with C       | -- | Gottfried Schaum's outline          |
| 4.C The complete Reference | -- | Herbert Schildt Tata McGraw Hill    |

**Suggested E-learning references**

1. [https://www.w3schools.com/c/c\\_intro.php?external\\_link=true](https://www.w3schools.com/c/c_intro.php?external_link=true)
2. <https://www.geeksforgeeks.org/c-programming-language/>
3. <https://www.programiz.com/c-programming>
4. <https://www.javatpoint.com/c-programming-language-tutorial>
5. <https://www.freecodecamp.org/news/the-c-programming-handbook-for-beginners/>
6. <https://www.tutorialspoint.com/cprogramming/index.htm>
7. <https://www.codecademy.com/catalog/language/c>
8. <https://www.codechef.com/learn/course/c>
9. <https://archive.nptel.ac.in/courses/106/104/106104128/>
10. <https://www.youtube.com/watch?v=3QiItnIWmOM>
11. <https://www.digimat.in/nptel/courses/video/106105171/L01.html>
12. <https://www.digimat.in/nptel/courses/video/106105171/L07.html>

## Suggested Learning Outcomes

Upon completion of the course, the student shall be able to

### UNIT–1: Computer Hardware and Operating System Basics

1. **Hardware Identification:** Identification of external ports (USB, HDMI, VGA, Audio, Ethernet) and internal components (Motherboard, RAM, CPU, HDD/SSD, SMPS).
2. **Peripherals Connection:** Demonstration of proper connection of monitor, keyboard, mouse, and power cables.
3. **Internal Assembly:** Identification of RAM slot, processor socket, and storage connections inside the CPU cabinet.
4. **BIOS Setup:** Entering BIOS/UEFI setup and changing boot priority (CD/DVD vs USB vs HDD).
5. **OS Installation:** Installation of Windows or Linux operating system.
6. **Disk Management:** Disk partitioning and formatting (Creating System drive and Data drive).
7. **File Management:** File and folder operations: Create, Rename, Copy, Move, Delete.
8. **System Utilities:** Usage of keyboard shortcuts (Ctrl+C, Ctrl+V, Alt+Tab) and basic system properties check.

### UNIT–2: Office Automation Tools

1. **Resume Creation:** Creation of a professional resume using MS Word (using templates/tables).
2. **Formatting Skills:** Formatting text using fonts, alignment, bullets, numbering, and inserting images.
3. **Official Correspondence:** Drafting an official Leave Letter or Permission Letter using MS Word.
4. **Page Layout:** Setting Page margins, Orientation, Headers, Footers, and performing Spell Check.
5. **Data Entry:** Creation of a Student Marksheet in MS Excel with at least 5 subjects.
6. **Excel Formulas:** Use of basic formulas: SUM, AVERAGE, MIN, MAX.
7. **Conditional Logic:** Use of IF function to determine Pass/Fail status based on marks.

8. **Bar Charts:** Creation of a Bar Chart for comparing marks of different students.
9. **Pie Charts:** Creation of a Pie Chart for result analysis (Pass percentage vs Fail percentage).
10. **Presentation Design:** Preparation of a PowerPoint presentation (minimum 5 slides) on a technical or general topic.
11. **Presentation Effects:** Applying slide transitions and custom animations to the presentation.

### UNIT-3: Introduction to C Programming

1. **First Program:** Program to print “Hello World” and Student Details (Name, Roll No).
2. **Debugging:** Program to demonstrate compilation process and correction of syntax errors.
3. **Arithmetic:** Program for Addition, Subtraction, Multiplication, and Division of two numbers.
4. **Geometry:** Program to calculate the Area and Circumference of a Circle ( $A=\pi r^2$ ).
5. **Swapping:** Program to swap two numbers using a third variable.
6. **Temperature:** Program to convert temperature from Celsius to Fahrenheit.
7. **Sign Check:** Program to check whether a number is Positive or Negative.
8. **Eligibility Check:** Program to check voting eligibility based on age (Using `if-else`).
9. **Comparison:** Program to find the largest of three numbers using nested `if-else`.
10. **Modulus Operator:** Program to check whether a number is Even or Odd.

### UNIT-4: Control Structures and Functions

1. **Looping (While):** Program to print natural numbers from 1 to N.
2. **Looping (For):** Program to print the multiplication table of a given number.
3. **Digit Logic:** Program to find the sum of digits of a number (e.g., 123→6).
4. **Reverse Logic:** Program to reverse a given number (e.g., 123→321).
5. **Palindrome (Number):** Program to check whether a number is a Palindrome.
6. **Prime Check:** Program to check whether a number is Prime.
7. **Series Generation:** Program to generate Fibonacci series up to N terms.

8. **Switch-Case:** Menu-driven Calculator (Add, Sub, Mul, Div) using `switch-case`.
9. **Switch-Logic:** Program to display the Day of the Week (1=Monday, etc.).
10. **Function (Math):** Program to calculate Factorial using a user-defined function.
11. **Function (Logic):** Program to check Even or Odd using a user-defined function.

#### **UNIT-5: Advanced C Programming**

1. **Array Basics:** Program to store elements in an array and find their Sum and Average.
2. **Search:** Program to find Maximum and Minimum elements in an array.
3. **Sorting:** Program to sort array elements in Ascending order (Bubble Sort).
4. **Matrix Addition:** Program to add two 3×3 matrices.
5. **Matrix Transpose:** Program to find the transpose of a matrix.
6. **String Stats:** Program to count vowels, consonants, and spaces in a string.
7. **String Palindrome:** Program to check whether a string is a Palindrome.
8. **String Logic:** Program to compare two strings without using the library function (`strcmp`).
9. **Structures:** Program using `struct` to store and display Student details (ID, Name, Marks).
10. **Pointer Basics:** Program demonstrating pointer declaration and dereferencing.
11. **Pointer Arithmetic:** Program to access array elements using pointers.
12. **Bitwise Logic:** Program to demonstrate `&` (AND), `|` (OR), `^` (XOR) operations.
13. **Bit Manipulation:** Program to **Set**, **Clear**, **Toggle**, and **Check** a specific bit of an integer.
14. **Embedded Concept:** Demonstration of relevance of Bitwise operations to LED ON/OFF control (Conceptual).

### Suggested Student Activities

1. **Know Your PC:** Group activity where students open a CPU cabinet, identify physical components (RAM, HDD, Processor), disconnect cables, and reassemble them to remove fear of hardware.
2. **OS Installation Drive:** Students learn to create a bootable USB drive (using tools like Rufus) and perform a clean installation of Windows or Linux on a lab computer.
3. **Professional Documentation:** Students create a "Class Newsletter" or "Event Brochure" using MS Word features like columns, headers, image wrapping, and tables.
4. **Real-World Data Analysis:** Collect real data (e.g., class test scores or grocery budget) and create an Excel sheet with Formulas, Conditional Formatting, and Graphs to visualize the data.
5. **Tech Talk Presentation:** Each student delivers a 5-minute PowerPoint presentation on a latest technical topic (e.g., 5G, AI, Electric Vehicles) to improve communication skills.
6. **Flowchart Design Challenge:** Before coding, students draw detailed flowcharts on chart paper for complex logic problems like "ATM Withdrawal Process" or "Traffic Light Control."
7. **Bug Hunter:** Students are given printed code sheets with intentional syntax/logical errors and must identify and correct them manually (Dry Run).
8. **C Mini-Project:** Students work in pairs to develop simple console-based applications like a "Quiz Game," "Simple Calculator," or "Student Record System" using Structures and Functions.

### CO-PO Mapping Matrix

|       | Basic and Discipline Specific Knowledge | Problem Analysis | Design/Development of Solutions | Engineering Tools, Experimentation and Testing | Engineering Practices for Society, Sustainability and Environment | Project Management | Lifelong Learning | Linked PO |
|-------|-----------------------------------------|------------------|---------------------------------|------------------------------------------------|-------------------------------------------------------------------|--------------------|-------------------|-----------|
| CO/PO | PO1                                     | PO2              | PO3                             | PO4                                            | PO5                                                               | PO6                | PO 7              |           |
| CO1   | 2                                       | 1                | 3                               | 3                                              | 1                                                                 |                    | 1                 | 1,2,3,4,7 |
| CO2   | 1                                       |                  | 3                               | 3                                              |                                                                   | 1                  | 2                 | 1,3,4,6,7 |
| CO3   | 3                                       | 3                | 3                               | 2                                              |                                                                   |                    | 1                 | 1,2,3,4,7 |
| CO4   | 3                                       | 3                | 3                               | 2                                              |                                                                   |                    | 1                 | 1,2,3,4,7 |
| CO5   | 2                                       | 3                | 3                               | 2                                              |                                                                   |                    | 2                 | 1,2,3,4,7 |

### Course Content and Blue Print of Marks for MID Exams

| Subject                                     | Hours/<br>Periods | Marks for MIDs |           |      | Marks<br>weightage | %Weightage |
|---------------------------------------------|-------------------|----------------|-----------|------|--------------------|------------|
|                                             |                   | Coding         | Execution | Viva |                    |            |
| Computer Fundamentals and C Programming Lab | 90                | 10             | 10        | -    | 20                 | 100        |
| <b>Total</b>                                | 90                | 20             |           |      | 20                 | 100        |

### Course Content and Blue Print of Marks for SEE

| Subject                                     | Hours/<br>Periods | Marks for SEE |           |      | Marks<br>weightage | %Weightage |
|---------------------------------------------|-------------------|---------------|-----------|------|--------------------|------------|
|                                             |                   | Coding        | Execution | Viva |                    |            |
| Computer Fundamentals and C Programming Lab | 90                | 20            | 10        | 10   | 40                 | 100        |
| <b>Total</b>                                | 90                | 40            |           |      | 40                 | 100        |