



# ANDHRA KESARI UNIVERSITY:: ONGOLE

Model Syllabus for Computer Science (Minor) in consonance with Curriculum framework w.e.f. AY 2025-26

## COURSE STRUCTURE

Year	Semester	Course	Title of the Course	No. of Hrs /Week	No. of Credits
II	III	1	Computer Fundamentals and Office Automation	3	3
			Computer Fundamentals and Office Automation-Practical	2	1
	IV	2	Problem Solving Using C	3	3
			Problem Solving Using C-Practical	2	1
III	V	3	Database Management System	3	3
			Database Management System-Practical	2	1
		4	OOPS Through JAVA	3	3
			OOPS Through JAVA-Practical	2	1
	VI	5	Python Programming	3	3
			Python Programming-Practical	2	1
		6	Web Interface Design Technologies	3	3
			Web Interface Design Technologies-Practical	2	1

Signatures of BOS Members

1) *[Signature]*  
10/09/2025

2) *[Signature]*  
10/09/2025

3) *[Signature]*  
10/09/2025  
U. Javala, DSR, IITM,  
Learning and

4) *[Signature]*  
10/09/2025





**ANDHRA PRADESH STATE COUNCIL OF HIGHER  
EDUCATION**

**Model Syllabus for Computer Science (Minor) in consonance with Curriculum  
framework w.e.f. AY 2025-26**

**COURSE STRUCTURE**

Year	Semester	Course	Title of the Course	No. of Hrs /Week	No. of Credits
II	III	1	Computer Fundamentals and Office Automation	3	3
			Computer Fundamentals and Office Automation-Practical	2	1
	IV	2	Problem Solving Using C	3	3
			Problem Solving Using C-Practical	2	1
III	V	3	Database Management System	3	3
			Database Management System-Practical	2	1
		4	OOPS Through JAVA	3	3
			OOPS Through JAVA-Practical	2	1
	VI	5	Python Programming	3	3
			Python Programming-Practical	2	1
		6	Web Interface Design Technologies	3	3
			Web Interface Design Technologies-Practical	2	1

Signatures of BoB Members

- 1) Ch. Prasad 10/9/2025 Dr. Ch. Prasad, Lecturer in CS, K.R.K. G.D.C.   
 *changed* Addank
- 2) Dr. Hanumanth Rao J. 10/09/25 Dr. Hanumanth Rao J., No.   
 Lecturer in Comp. Sci.,   
 T.R.G.D.C(A), Rameswaram
- 3) V. Sarala 10/9/25 V. Sarala, Lecturer in CS, DSRDWC, Ongole
- 4) V. Saritha Reddy 10/09/25 V. Saritha Reddy, Lect. in CS, GCV(A), Guntur

## SEMESTER-III

### COURSE 1: COMPUTER FUNDAMENTALS AND OFFICE AUTOMATION

Theory

Credits: 3

3 hrs/week

---

#### Course Objectives

1. Understand foundational computing concepts, including number systems, the evolution of computers, block diagrams, and generational progress.
2. Develop knowledge of computer architecture, focusing on system organization and networking fundamentals.
3. Acquire practical skills in document creation, formatting, and digital presentations using word processing tools.
4. Gain proficiency in spreadsheet operations, such as data entry, formulas, functions, and charting techniques.
5. Introduce data visualization and basic modelling principles, fostering analytical thinking in structuring and interpreting data sets.

#### Course Outcomes

1. At the End of the Course, The Students will be able to explain different number systems, the historical evolution of computers, and identify key components in a block diagram.
2. Learners will demonstrate basic blocks of a computer and fundamental networking knowledge.
3. Learners will create professional-level documents and design visually appealing presentations using word processing software and presentation software.
4. Learners will manipulate data within spreadsheets, apply formulas, and generate accurate summaries and visualizations.
5. Learners will apply data modelling techniques to analyze, organize, and represent data effectively in various scenarios.

#### Unit 1. Number Systems, Evolution , Block Diagram and Generations:

④ Number Systems: Binary, Decimal, Octal, Hexadecimal; conversions between number systems.

① Evolution of Computers: History from early mechanical devices to modern-day systems.

③ Block Diagram of a Computer: Components like Input Unit, Output Unit, Memory, CPU (ALU + CU).

② Generations of Computers: First to Fifth Generation – technologies, characteristics, examples.

#### Unit 2. Basic organisation and N/W fundamentals:

Computer Organization: Functional components – Input/Output devices, Storage types, Memory hierarchy.

Types of Computers: Micro, Mini, Mainframe, and Supercomputers.

Change  
The  
order  
A  
to  
Numbers

**Networking Fundamentals:** Definition, need for networks, types (LAN, WAN, MAN), topology (Star, Ring, Bus).

**Internet Basics:** IP Address, Domain Name, Web Browser, Email, WWW.

**Unit 3. Word Processing and presentations:**

**Word Processing Basics:** Using MS Word/Google Docs – formatting, styles, tables, mail merge.

**Presentation Tools:** Using PowerPoint/Google Slides – slide design, animations, transitions.

**Applications:** Creating resumes, reports, brochures, and presentations.

**Keyboard Shortcuts**

**Unit 4. Spreadsheet Basics:**

**Spreadsheet Concepts:** Understanding rows, columns, cells in tools like MS Excel/Google Sheets.

**Functions and Formulas:** SUM, AVERAGE, IF, COUNT.

**Charts and Graphs:** Creating visual representations

**Data Handling:** Sorting, filtering, conditional formatting.

**Text Functions:** LEFT, RIGHT, MID, LEN, TRIM, CONCAT, TEXTJOIN

**Advanced Functions: Logical:** IF, AND, OR, IFERROR, **Lookup:** VLOOKUP, HLOOKUP, XLOOKUP, INDEX, MATCH

**Unit 5. Data Modelling:**

**Conditional Formatting:** Custom rules, Color scales, Icon sets, Data bars

**Data Analysis Tools:** Pivot Tables and Pivot Charts, Data Validation (Drop-downs, Input Messages, Error Alerts), What-If Analysis: Goal Seek, Scenario Manager, Data Tables

**Charts and Dashboards:** Creating Interactive Dashboards, Using slicers with Pivot Tables, Combo Charts and Sparklines

**Productivity Tips:** Using Named Ranges, Freeze Panes, Split View

*Remove from  
Syllabus*

**Textbooks:**

1. Fundamentals of Computers, Reema Thareja, Oxford University Press, 2nd edition
2. Fundamentals of Computers, V. Rajaraman – PHI Learning
3. Introduction to Computers by Peter Norton – McGraw Hill
4. Microsoft Office 365 In Practice by Randy Nordell – McGraw Hill Education

## References:

1. Excel 2021 Bible by Michael Alexander, Richard Kusleika – Wiley
2. Networking All-in-One For Dummies by Doug Lowe – Wiley
3. Microsoft Official Docs and Training: <https://learn.microsoft.com>
4. Google Workspace Learning Center: <https://support.google.com/a/users/>

## Activities:

### Unit 1: Number Systems & Computer Evolution

**Outcome:** At the End of the Course, The Students will be able to explain different number systems, the historical evolution of computers, and identify key components in a block diagram.

**Activity:** Create a digital poster or infographic comparing number systems (binary, decimal, octal, hexadecimal) and illustrating the timeline of computer generations with key innovations.

**Evaluation Method:** Rubric-based assessment of the poster presentation on a 10-point scale focusing on:

- Accuracy of number system conversions
- Correct identification of block diagram components
- Visual organization and creativity

### Unit 2: Computer Architecture & Networking Basics

**Outcome:** Learners will demonstrate basic blocks of a computer and fundamental networking knowledge.

**Activity:** Design a concept map showing the internal architecture of a computer and types of networks (LAN, WAN, MAN), including devices and topologies.

**Evaluation Method:** Checklist-based peer review and instructor validation:

- Completeness of the map
- Correctness of networking concepts
- Use of appropriate terminology
- Logical flow and structure of the map

### Unit 3: Word Processing & Presentation Design

**Outcome:** Learners will create professional-level documents and design visually appealing presentations using word processing software and presentation software.

**Activity:** Prepare a formal report (e.g., project proposal) in a word processor and present it using a slide deck with transitions, embedded media, and design elements.

**Evaluation Method:** Performance-based evaluation using a 10-point scoring scale:

- Formatting and structure of the document
- Presentation aesthetics and clarity
- Communication skills during presentation

#### **Unit 4: Spreadsheet Analysis & Visualization**

**Outcome:** Learners will manipulate data within spreadsheets, apply formulas, and generate accurate summaries and visualizations.

**Activity:** Analyze a dataset (e.g., student scores or sales data) using spreadsheet software. Apply formulas (SUM, AVERAGE, IF, VLOOKUP) and create relevant charts.

**Evaluation Method:** Practical test with a rubric:

- Correct use of formulas
- Accuracy of data summaries

#### **Unit 5: Data Analysis and Visualization:**

**Outcome:** Learners will apply data modelling techniques to analyze, organize, and represent data effectively in various scenarios.

**Activity:** Prepare an interactive dashboard for a given data set using EXCEL.

**Evaluation Method:** Evaluation of the dashboard on a 10-point scoring scale:

- Presentation aesthetics and clarity
- Interactiveness
- Communication skills during presentation

## SEMESTER-III

### COURSE 1: COMPUTER FUNDAMENTALS AND OFFICE AUTOMATION

Practical

Credits: 1

2 hrs/week

---

#### List of Experiments:

1. Demonstration of Assembling and Dessembling of Computer Systems.
2. Identify and prepare notes on the type of Network topology of your institution.
3. Prepare your resume in Word.
4. Using Word, write a letter to your higher official seeking 10-days leave.
5. Prepare a presentation that contains text, audio and video.
6. Using a spreadsheet, prepare your class Time Table.
7. Using a Spreadsheet, calculate the Gross and Net salary of employees (Min 5) considering all the allowances.
8. Generate the class-wise and subject-wise results for a class of 20 students. Also generate the highest and lowest marks in each subject.
9. Using IF, AND, OR, and IFERROR to Automate Grade Evaluation.
  - a. Create a table of student scores in different subjects.
  - b. Use IF to assign grades (A/B/C/Fail).
  - c. Use IFERROR to handle missing scores or invalid data.
10. *Employee Database Search Using VLOOKUP, HLOOKUP, XLOOKUP, INDEX, and MATCH*
  - a. Create a database of employees (Name, ID, Department, Salary).
  - b. Implement VLOOKUP to search by employee ID.
  - c. Use HLOOKUP to extract department heads by role.
  - d. Apply XLOOKUP for more flexible searches.
  - e. Use INDEX + MATCH as an alternative to VLOOKUP.
11. Sales Report Analysis Using Pivot Tables and Charts
  - a. Use a dataset of product sales (Product, Region, Date, Quantity, Revenue).
  - b. Create Pivot Tables to summarize data by region/product.
  - c. Insert Pivot Charts for visual analysis (e.g., bar, line).
  - d. Add slicers to make the dashboard interactive.
12. Designing a Data Entry Form with Drop-downs and Input Rules
  - a. Create a student registration form.
  - b. Add drop-down lists for course selection using Data Validation.
  - c. Add input messages to guide users.
  - d. Add error alerts for wrong entries.
13. Monthly Budget Planning using Goal Seek and Scenario Manager
  - a. Create a simple personal budget (income, expenses, savings).
  - b. Use Goal Seek to determine income needed to save a desired amount.
  - c. Use Scenario Manager to compare different budgeting scenarios (best/ worst/ realistic case).
  - d. Create a one-variable Data Table to analyze how different expenses affect savings.

#### 14. Dashboard Creation Using Combo Charts, Sparklines & Slicers

- a. Use existing sales or attendance data.
- b. Insert combo charts (e.g., column + line).
- c. Add sparklines to show trends.
- d. Use slicers with Pivot Tables to control dashboard elements.
- e. Finalize and format for interactivity.

## SEMESTER-IV

### COURSE 2: PROBLEM SOLVING USING C

Theory

Credits: 3

3 hrs/week

---

#### Course Objectives:

1. Understand the fundamentals of computer programming, Apply structured problem-solving approaches using algorithms, flowcharts, and C programming constructs.
2. Develop efficient logic using decision-making, loop, and jump control statements.
3. Utilize derived data types like arrays and strings for modular program design.
4. Design and implement modular solutions using functions, recursive logic, pointer operations, and dynamic memory management.
5. Handle complex data structures including structures, unions, and text file operations.

#### Course Outcomes:

At the End of the Course, The Students will be able to:

1. Understand basic computing concepts, programming paradigms and write structured C programs.
2. Apply control flow statements to solve logical and repetitive tasks in C.
3. Implement arrays and string operations to manage and manipulate data efficiently.
4. Design modular code using functions, recursion, and appropriate parameter passing.
5. Utilize pointers and memory operations for effective data handling. Demonstrate competence in dynamic memory allocation and text file processing.

#### Unit 1. Introduction to computer programming:

Introduction, Types of software, Compiler and interpreter, Concepts of Machine level, Assembly level and high-level programming, Flowcharts and Algorithms, Fundamentals of C: History of C, Features of C, C Tokens-variables and keywords and identifiers, constants and Data types, Rules for constructing variable names, Operators, Structure of C program, Input /output statements in C-Formatted and Unformatted I/O

#### Unit 2. Control statements:

Decision making statements: if, if else, else if ladder, switch statements. Loop control statements: while loop, for loop and do-while loop. Jump Control statements: break, continue and goto.

#### Unit 3. Derived data types in C:

Arrays: One Dimensional arrays - Declaration, Initialization and Memory representation; Two Dimensional arrays -Declaration, Initialization and Memory representation. Strings: Declaring & Initializing string variables; String handling functions, Character handling functions

#### Unit 4. ~~Functions~~ *Pointers and Functions*

Pointers: Pointer data type, Pointer declaration, initialization, accessing values using pointers. Pointer arithmetic, Pointers and arrays. ↵

Function Prototype, definition and calling. Return statement. Nesting of functions. Categories of functions. Recursion (Basic Concept only). Parameter Passing by address & by value. Local and Global variables. Storage classes: automatic, external, static and register.

#### Unit 5. ~~Dynamic Memory Management~~ *Structures, unions and File management*

Introduction, Functions-malloc, calloc, realloc, free Structures: Basics of structure, structure members, accessing structure members, nested structures, array of structures, structure and functions, structures and pointers. Unions - Union definition; difference between Structures and Unions. Working with text files - modes: opening, reading, writing and closing text files.

#### Text Books:

1. Programming in ANSI C, E. Balagurusamy, Tata McGraw Hill, 6 th Edn,
2. Computer fundamentals and programming in C, Reema Theraja, Oxford University Press

#### Reference Books:

1. Let us C, Y Kanetkar, BPB publications
2. Head First C: A Brain-Friendly Guide, David Griffiths, Dawn Griffiths

#### Activities:

**Outcome:** Understand basic computing concepts, programming paradigms and write structured C programs.

**Activity:** Create a concept map of computing fundamentals and programming paradigms (procedural, structured, object-oriented). Then, they write a structured C program (e.g., a calculator or student grade system) using proper syntax, indentation, and modular design.

**Evaluation Method:** Rubric-based Code Review & Viva to check the

- The correctness of the concept map
- Correct use of structure (main + functions)
- Identification of paradigm used
- Code readability and documentation

**Outcome:** Apply control flow statements to solve logical and repetitive tasks in C.

**Activity:** Implement a program that solves a logic puzzle (e.g., number guessing game, pattern generation, or prime number finder) using if, switch, for, while, and do-while.

**Evaluation Method:** Automated Test Cases + Peer Review to check the

- Correct use of control statements
- Logical correctness of output

- Efficiency and edge case handling
- Peer feedback on clarity and logic

**Outcome:** Implement arrays and string operations to manage and manipulate data efficiently.

**Activity:** Build a program that stores and arranges student marks in ascending and descending order using arrays and performs string operations like concatenation, comparing, and formatting names.

**Evaluation Method:** Functional Demonstration + Code Walkthrough to check the

- Correct array and string usage
- Memory efficiency
- Handling of invalid inputs
- Explanation of sorting/searching logic

**Activity:**

- **Recursive Problem Solver**

Students write a modular program to solve a recursive problem (e.g., factorial, Fibonacci, or Tower of Hanoi) using functions with parameters and return values.

**Evaluation Method:**

- **Code Trace + Written Quiz**

- Correct function decomposition
- Proper parameter passing (by value/reference)
- Recursion depth and base case handling
- Quiz on tracing recursive calls

**Outcome:** Utilize pointers and memory operations for effective data handling. Demonstrate competence in dynamic memory allocation and text file processing.

**Activity:** Create a program that dynamically stores user input (e.g., survey responses) using pointers and writes/reads the data to/from a text file.

**Evaluation Method:** Memory Debugging + File I/O Assessment to check the

- Proper use of malloc, calloc, realloc, and free
- Pointer arithmetic and dereferencing
- File creation, reading, writing, and error handling
- Use of tools like Valgrind or manual memory trace (Optional for Unix flavours)

SEMESTER-IV

*No change*

COURSE 2: PROBLEM SOLVING USING C

Practical

Credits: 1

2 hrs/week

---

List of Experiments:

1. Write a program to check whether the given number is Armstrong or not.
2. Write a program to find the sum of individual digits of a positive integer.
3. Write a program to generate the first n terms of the Fibonacci sequence.
4. Write a program to find both the largest and smallest number in a list of integer values
5. Write a program to demonstrate change in parameter values while swapping two integer variables using Call by Value & Call by Address
6. Write a program to perform various string operations.
7. Write a program to search an element in a given list of values.
8. Write a program that uses functions to add two matrices.
9. Write a program to calculate factorial of given integer value using recursive functions
10. Write a program for multiplication of two N X N matrices.
11. Write a program to sort a given list of integers in ascending order.
12. Write a program to calculate the salaries of all employees using Employee (ID, Name, Designation, Basic Pay, DA, HRA, Gross Salary, Deduction, Net Salary) structure.
  - a. DA is 30 % of Basic Pay
  - b. HRA is 15% of Basic Pay
  - c. Deduction is 10% of (Basic Pay + DA)
  - d. Gross Salary = Basic Pay + DA+ HRA
  - e. Net Salary = Gross Salary - Deduction
13. Write a program to read / write the data from / to a file.
14. Write a program to reverse the contents of a file and store in another file.
15. Write a program to create Book (ISBN, Title, Author, Price, Pages, Publisher) structure and store book details in a file and perform the following operations
  - a. Add book details
  - b. Search a book details for a given ISBN and display book details, if available
  - c. Update a book details using ISBN
  - d. Delete book details for a given ISBN and display list of remaining Books

No change

## SEMESTER-V

### COURSE 3: DATABASE MANAGEMENT SYSTEMS

Theory

Credits: 3

3 hrs/week

---

#### Course Objectives:

1. To understand the fundamentals of data, information, and the evolution from file-based systems to modern database management systems.
2. To develop the ability to design conceptual data models using Entity-Relationship (ER) and Enhanced ER diagrams.
3. To explore relational model principles, such as keys, integrity constraints, relational algebra and calculus, and normalization.
4. To perform data definition and manipulation using SQL commands including queries, joins, subqueries, views, and set operations.
5. To apply procedural logic using PL/SQL, incorporating control structures, functions, procedures, and database triggers.

#### Course Outcomes:

At the End of the Course, The Students will be able to:

1. **Describe** the fundamentals of data, database systems, and the differences between file-based and database approaches. **Compare and classify** various DBMS architectures, data models, and their components, including the three-schema architecture.
2. **Design** conceptual data models using Entity-Relationship and Enhanced ER diagrams, applying generalization, specialization, and constraints.
3. **Apply** relational model concepts, including CODD rules, relational algebra, relational calculus, and normalization techniques.
4. **Construct and execute** SQL queries for data definition, manipulation, aggregation, joining, and subqueries, including views and set operations.
5. **Develop** PL/SQL programs incorporating control structures, procedures, functions, and triggers to manage database behavior effectively.

#### Unit 1. Overview of Database Management System:

Introduction to data, information, database, database management systems, file-based system, Drawbacks of file-Based System, database approach, Classification of Database Management Systems, advantages of database approach, Various Data Models, Components of Database Management System, three schema architecture of data base, costs and risks of database approach.

#### Unit 2. Entity-Relationship Model:

Introduction, the building blocks of an entity relationship diagram, classification of entity sets, attribute classification, relationship degree, relationship classification, reducing ER diagram to tables, enhanced entity-relationship model (EER model), generalization and specialization, IS A relationship and attribute inheritance, multiple inheritance, constraints on specialization and generalization, advantages of ER modeling.