Complete Roadmap for C Programming with Project.

-SolitaryHorkos

C Programming Roadmap – Intermediate Level

🎯 Goals:

- ✅ Learn advanced concepts such as arrays, strings, and functions.
- ✅ Understand pointers and memory management in depth.
- ✅ Explore file handling techniques.
- ✅ Work with structures, unions, and enumerations.
- ✅ Implement recursion and basic algorithms.
- ✅ Develop modular programming skills using multiple files.

🔰 Phase 1: Advanced Arrays & Strings

📌 Arrays

- 🔹 One-dimensional and multi-dimensional arrays.
- 🔹 Working with arrays of structures.

📌 Strings and String Manipulation

- 🔹 Character arrays vs. pointers.
- 🔹 String functions: strlen(), strcpy(), strcat(), strcmp().

🔰 Phase 2: Pointers & Memory Management

📌 Understanding Pointers

- 🔹 Pointer basics and pointer arithmetic.
- 🔹 Pointers and arrays relationship.
- 🔹 Function pointers.

📌 Dynamic Memory Allocation

- 🔹 Using malloc(), calloc(), realloc(), and free() for memory management.

Complete Roadmap for C Programming with Project.

-SolitaryHorkos

🔰 Phase 3: Structures, Unions & Enumerations

📌 Structures and Unions

- 🔹 Defining and using struct and union.
- 🔹 Nested structures.
- 🔹 Arrays of structures.

📌 Enumerations

- 🔹 enum and its applications.

🔰 Phase 4: File Handling & Advanced I/O

📌 Working with Files

- 🔹 File operations: fopen(), fwrite(), fread(), fclose().
- 🔹 File modes (r, w, a, etc.).
- 🔹 Reading and writing structured data to files.
- 🔹 Error handling in file operations.

🔰 Phase 5: Recursion & Algorithm Basics

📌 Recursion and Backtracking

- 🔹 Understanding recursion and writing recursive functions.
- 🔹 Common recursive problems: Factorial, Fibonacci, GCD, Tower of Hanoi.

📌 Sorting and Searching Algorithms

- 🔹 Sorting: Bubble sort, selection sort, insertion sort.
- 🔹 Searching: Linear search, binary search.

Complete Roadmap for C Programming with Project.

-SolitaryHorkos

🔰 Phase 6: Modular Programming & Preprocessor Directives

📌 Multi-file Programming
- 🔹 Organizing code into multiple files.
- 🔹 Using header files and modular programming concepts.

📌 Preprocessor Directives
- 🔹 Understanding #define, #include, #ifdef, #ifndef.
- 🔹 Writing macros and using conditional compilation.

📌 Command-line Arguments
- 🔹 Using argc and argv to accept user input from the command line.

🔰 Phase 7: Introduction to Data Structures

📌 Fundamental Data Structures
- 🔹 Introduction to linked lists, stacks, and queues.
- 🔹 Basic operations such as insertion, deletion, and traversal.

🔰 Phase 8: Debugging & Error Handling

📌 Debugging Tools
- 🔹 Using gdb for debugging C programs.
- 🔹 Understanding compiler warnings and runtime errors.

🏆 Skills to Master by the End of This Phase:

✅ Working with pointers and dynamic memory allocation.
✅ Manipulating strings and handling files efficiently.

Complete Roadmap for C Programming with Project.

-SolitaryHorkos

- ✅ Writing modular and well-structured code.
- ✅ Implementing recursion and solving basic algorithmic problems.
- ✅ Understanding and applying structures, unions, and enumerations.
- ✅ Using preprocessor directives effectively.
- ✅ Debugging and optimizing C programs.

🎯 PROJECTS:

- **Student Management System**: Store and manage student records using file I/O and structures, performing basic CRUD operations.
- **Tic-Tac-Toe Game**: A two-player game using a 2D array, with a simple AI.
- **Simple File Encryption & Decryption**: Encode and decode text in a file.
- **Bank Management System**: Store user account information and allow basic transactions like deposits and withdrawals.
- **Library Management System**: Manage book records, including issuing and returning books.
- **Hangman Game**: A word-guessing game using strings, reading words from a file.
- **Calendar Application:** Display a calendar for any given year based on user input.
- **Snake Game**: Implement a simple version of the classic Snake game.
- **Basic Text Editor**: Read, write, and modify text files.
- **Employee Management System**: Store and update employee details.
- **Contact Book**: Store and manage contacts with name, phone, and email.
- **Simple Database System**: Store and retrieve data using files.
- **Word Counter**: Count words, lines, and characters in a text file.
- **Matrix Operations**: Perform addition, subtraction, and multiplication on matrices.
- **Password Manager**: Store and retrieve passwords securely.

Complete Roadmap for C Programming with Project.

-SolitaryHorkos

- **Expense Tracker** – Track daily expenses and generate reports using file I/O.
- **Text-based Address Book** – Manage contacts with options to add, delete, search, and update entries.
- **Basic Tetris Clone** – Implement a simplified version of Tetris, managing game states with arrays.
- **Linked List Library** – Build a custom library for linked lists, including operations like search and deletion.
- **Mini Command-Line Shell** – Write a basic shell to interpret simple commands.
- **Maze Solver** – Use recursion or depth-first search (DFS) to solve a maze represented in a 2D array.
- **Budget Tracker** – A simple expense recording system using file I/O.
- **Memory Visualizer** – Simulate dynamic memory allocation by showing how memory is allocated and freed in real time.
- **String Manipulation Library** – Create custom string functions.
- **Dynamic Array Implementation** – Implement a dynamic array using pointers.
- **Simple Inventory System** – Manage a list of items using structures and files.