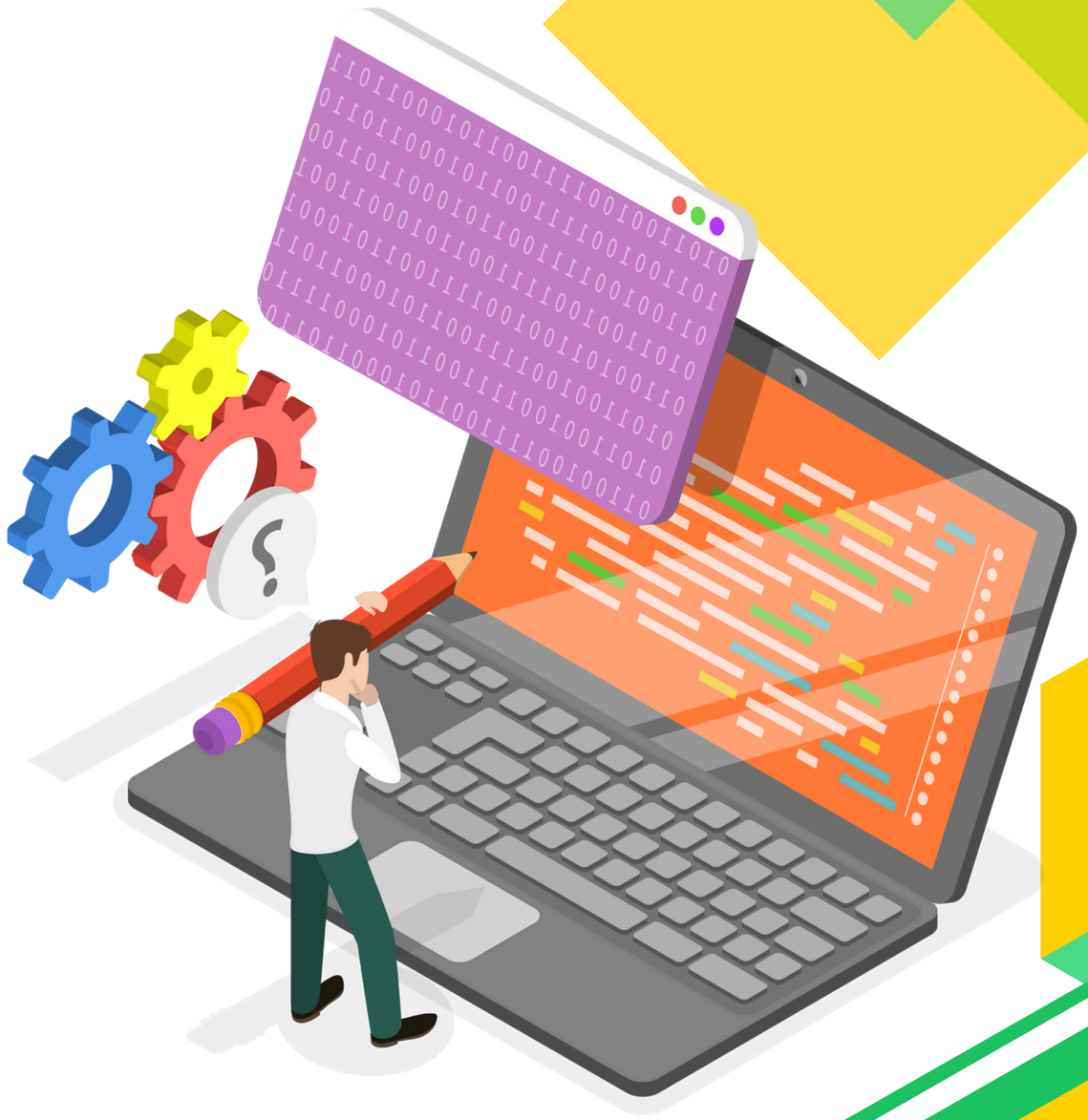
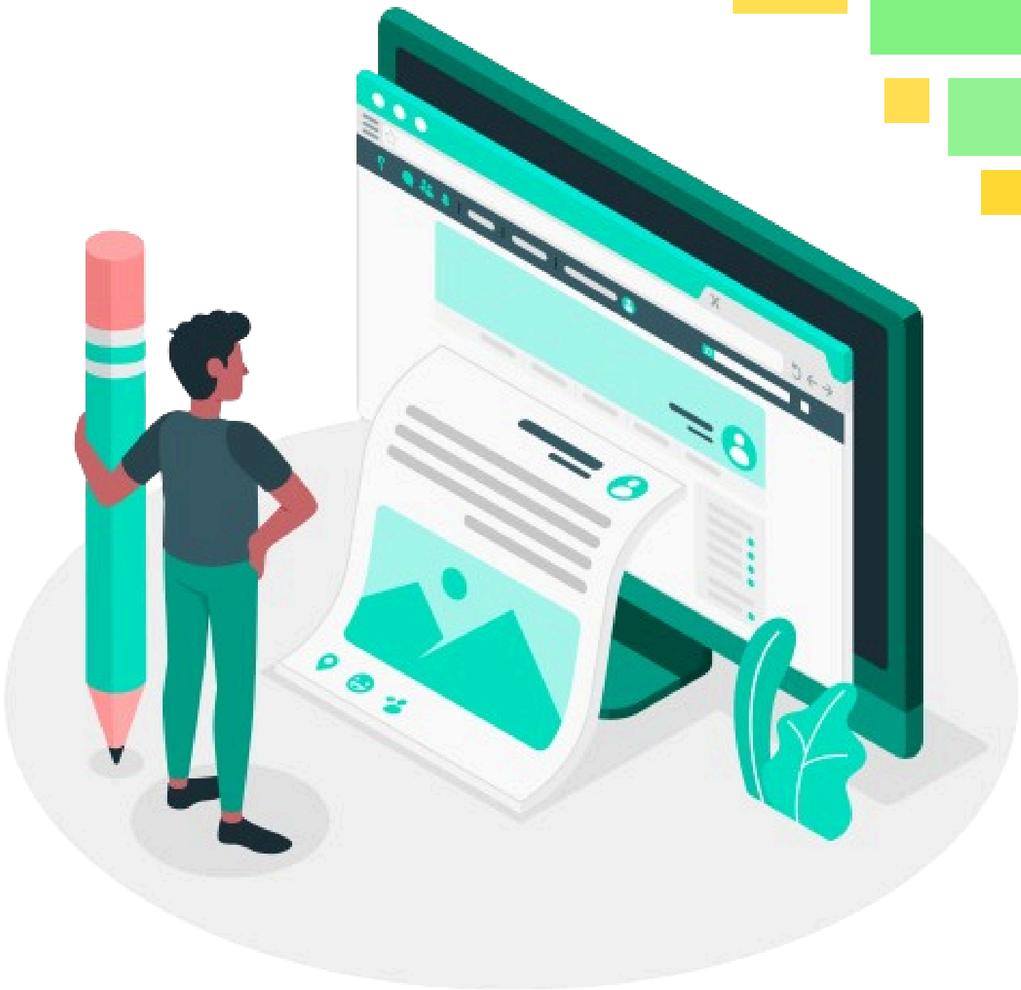
Codveda

C/C++ Development



Content

1. About Us
2. Instructions
3. Submission
4. Task List



About Us

Codveda

Welcome to Codveda Technology, where innovation meets excellence. Founded with a vision to empower businesses through cutting-edge IT solutions, we specialize in delivering tailored services that drive success in the digital era.

At Codveda, we offer a diverse range of services, including web development, app development, digital marketing, SEO optimization, AI/ML automation, and data analysis.

Our team of skilled professionals is committed to helping businesses unlock their full potential by providing innovative, scalable, and reliable solutions.

INSTRUCTIONS

Codveda

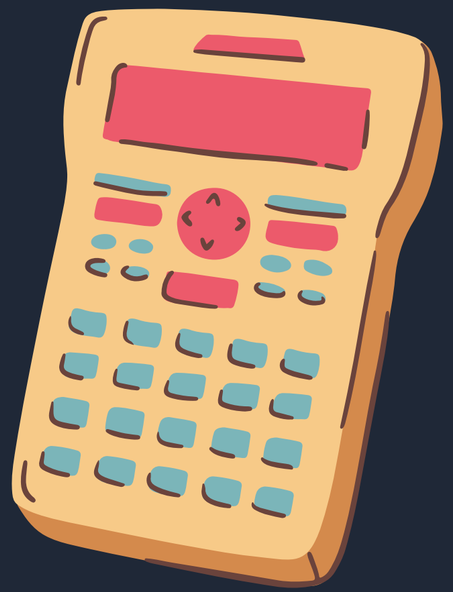
- Update your LinkedIn profile with your achievements, including the offer letter and completion certificate. Mention and tag @Codveda in your posts.
- Use hashtags like #CodvedaJourney, #CodvedaExperience, and #FutureWithCodveda to showcase your progress and experiences.
- Share your project completion updates on LinkedIn, accompanied by a video explanation and the GitHub project repository link.
- You will be provided with four tasks. Select and complete any three tasks within your domain to fulfill the internship requirements.
- Submit your completed tasks via the Codveda submission form. Ensure all tasks are submitted within the allocated 15-day period.

SUBMISSION

Codveda

- Create a professional video showcasing your internship projects and achievements.
- Host the video on LinkedIn to provide proof of your work and establish credibility among your peers. Consider tagging Codveda Technology in your posts to ensure they are notified of your work using hashtags like #CodvedaAchievements and #CodvedaProjects.
- A SUBMISSION FORM will be shared later. Till then, please continue your tasks and maintain a separate file for each level.
- When posting the video on LinkedIn, include engaging content that highlights your contributions and skills. Tailor the post to your specific internship domain to maximize impact and visibility.

Level 1 (Basic)



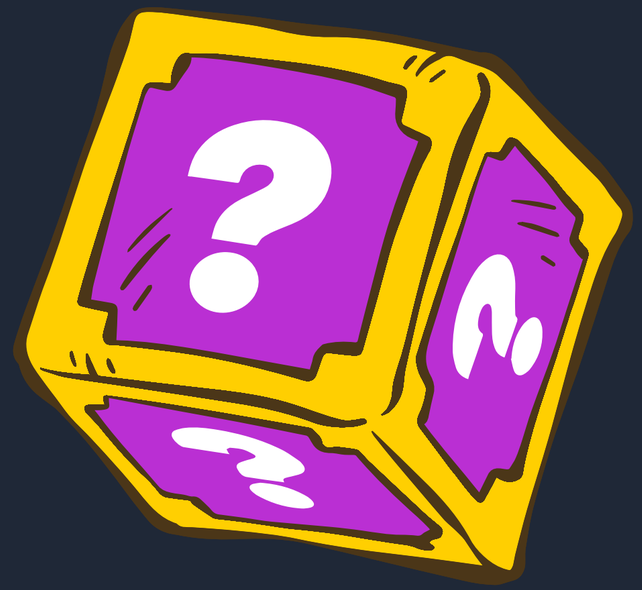
Task 1: Basic Calculator

- Description: Create a simple calculator program that performs basic arithmetic operations (addition, subtraction, multiplication, division).

Objectives:

- Implement user input and output functions.
- Handle basic operations using conditional statements or a switch case.
- Ensure proper handling of division by zero.
- Skills Covered: Input/output, conditional logic, arithmetic operations.

Level 1
(Basic)



Task 2: Number Guessing Game

- Description: Build a number guessing game where the program generates a random number, and the user tries to guess it.

Objectives:

- Use the rand() function to generate random numbers.
- Implement a loop to allow multiple attempts.
- Provide feedback on whether the guess was too high or too low.
- Skills Covered: Random number generation, loops, conditional logic.

Level 1 (Basic)

Task 3: String Manipulation

- Description: Write a program to implement basic string operations such as reversing a string, counting vowels, and checking if the string is a palindrome.

Objectives:

- Use character arrays and string functions.
- Implement custom functions for reversing and checking properties of strings.
- Skills Covered: Character arrays, loops, functions, string operations.

Level 2 (Intermediate)

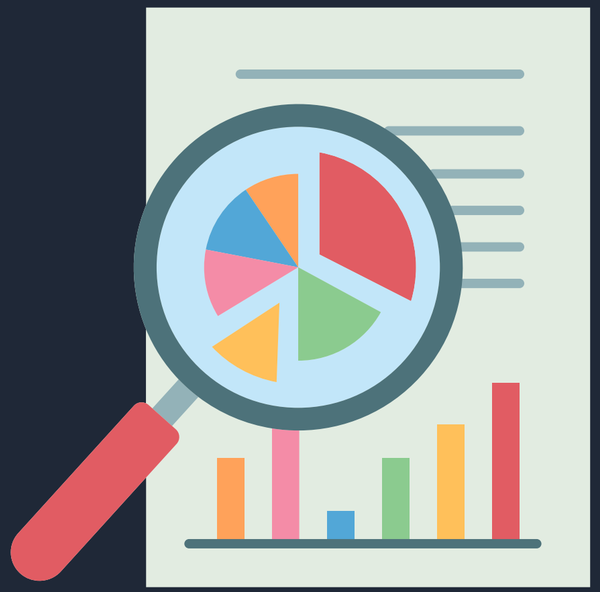
Task 1: Simple Student Management System

- Description: Create a console-based program to manage student records, including adding, deleting, and displaying student information.

Objectives:

- Use structs to represent student data (name, ID, grades).
- Implement file handling to save and load records.
- Implement menu-driven options for CRUD (Create, Read, Update, Delete) operations.
- Skills Covered: Structs, file I/O, arrays, pointers.

Level 2 (Intermediate)



Task 2: Sorting and Searching Algorithms

- Description: Implement popular sorting algorithms (e.g., bubble sort, quicksort) and searching algorithms (e.g., linear search, binary search).

Objectives:

- Write functions for sorting and searching arrays.
- Compare the time complexity of each algorithm.
- Integrate user input for testing different array sizes and data.
- Skills Covered: Algorithms, time complexity, functions, arrays.

Level 2 (Intermediate)

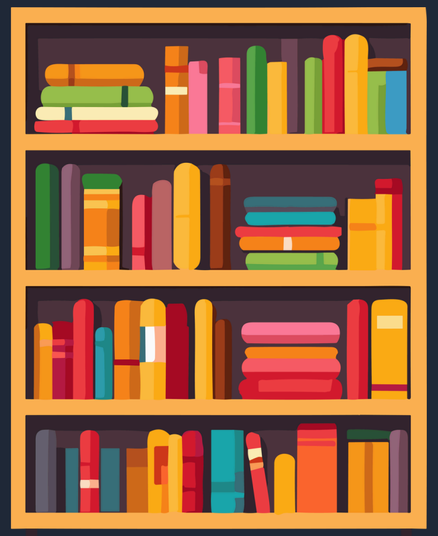
Task 3: Basic Data Structure Implementation

- Description: Implement data structures such as stacks or queues using arrays or linked lists.

Objectives:

- Implement push, pop, and display functions for stacks.
- Implement enqueue, dequeue, and display functions for queues.
- Skills Covered: Data structures, dynamic memory allocation, linked lists.

Level 3 (Advanced)



Task 1: Simple File System Simulation

- Description: Create a simple simulation of a file system to perform operations like creating, deleting, and reading files.

Objectives:

- Use file handling functions to simulate file operations.
- Create a menu-driven program with options for file operations.
- Implement error handling for invalid inputs or operations.
- Skills Covered: File I/O, structs, error handling, pointers.

Level 3 (Advanced)

Task 2: Multithreaded Application

- Description: Build a multithreaded program that simulates a simple producer-consumer problem.

Objectives:

- Use POSIX threads (pthreads) to create producer and consumer threads.
- Implement synchronization using mutexes and condition variables.
- Ensure threads operate in a safe and synchronized manner.

Level 3 (Advanced)

Task 3: Mini Game Development (e.g., Tic-Tac-Toe)

- Description: Develop a simple console-based Tic-Tac-Toe game for two players.

Objectives:

- Implement game logic and display the game board.
- Check for winning conditions and handle draws.
- Implement user input and turn switching between players.

How to Contact Us?

For additional information, kindly get in touch with our team.

 @codveda

 support@technofyz.com

 www.codveda.com