1D Array Practice Problems in C

Amaan Rahman

Introduction

This document contains a set of problems designed to help you master **1D arrays in C**. The problems are arranged from **basic** to **intermediate** to **advanced**. Try solving them in order and without looking at solutions initially.

Level 1: Basics

B1. Write a program to read an array of n integers from the user and print all the elements.

B2. Find the **sum** and **average** of all elements in a 1D array.

B3. Given an array, print the **maximum** and **minimum** elements.

B4. Count the total number of **even** and **odd** elements in an array.

B5. Reverse the elements of a given array and print the reversed array.

Level 2: Intermediate

I1. Given an array of integers, find the **second largest** and **second smallest** elements.

I2. Check whether a given element exists in the array. If yes, print its index; otherwise, print "Not Found".

I3. Count the frequency of each unique element in the array.

I4. Sort the array in **ascending** and **descending** order without using built-in functions.

I5. Merge two sorted arrays into a single sorted array.

Level 3: Advanced

A1. Rotate an array by k positions to the right without using an extra array.

A2. Move all zeros in the array to the end while maintaining the order of non-zero elements.

A3. Given a sorted array, remove all duplicate elements **in-place** and print the new length.

A4. Find the **majority element** (the element that appears more than $\frac{n}{2}$ times) if it exists.

A5. Implement a program to find the **maximum sum of a contiguous subarray** (Kadane's Algorithm).

Tips for Practice

- Try solving each problem **without using extra arrays** first.
- Focus on **loop control**, **indexing**, and **memory access**.
- Add input validation and test your code with edge cases.