Assignment 6

Topics: Function, Array and Pointers - II

Section A6.1: [Functions, Arrays, and Pointers]

6.1a: Write a C program to calculate the greatest common divisor (GCD) of two integers using a function. Call the function from 'main()' and print the result.

6.1b: Implement a function that takes an array and its size as arguments and returns the sum of all its elements. Use it in 'main()'.

6.1c: Write a program that reads 10 integers into an array, calculates the sum of even numbers, and prints the result.

6.1d: Write a C program that swaps the values of two variables using pointers (without direct swapping in 'main()').

6.1e: Create a program that sorts an array of size 10 (entered by the user) in ascending order and prints the sorted elements.

6.1f: Write a function that increments an integer by 1 using pointers. Call it in 'main()' and display the updated value.

6.1g [Bonus]: Write a function to check if a number is a palindrome. Use the function in 'main()' to check user-entered numbers.

Section A6.2: [Array with Pointers]

6.2a: Write a program to traverse and print an array of integers using pointer arithmetic instead of array indexing.

6.2b: Create a C program to reverse an array of 10 integers using a pointer to access the elements. Display the reversed array.

6.2c: Implement a C program that uses pointers to find the largest and smallest elements in an array of integers.

6.2d: Write a program that copies the contents of one array to another using pointers.

6.2e [Bonus]: Create a function that accepts an array and its size as arguments and counts how many elements are greater than a given number (taken from the user). Use pointers in your implementation and print the result in 'main()'.

Section A6.3: [Array with Functions]

6.3a: Write a function that takes an array and its size as arguments and returns the average of all the elements. Call the function from 'main()'.

6.3b: Create a program where a function takes an array and its size, reverses the array, and prints the reversed array in 'main()'.

6.3c: Write a function that sorts an array in descending order. Pass the array to the function from 'main()' and print the sorted array.

6.3d: Implement a function that takes an array and finds the second-largest element. Call the function from 'main()' and print the result.

6.3e [Bonus]: Write a function that receives an array and its size, removes all duplicate elements from the array, and returns the new size of the array. Call the function from 'main()' and print the modified array.

Section A6.4: [Function with Pointers]

6.4a: Write a function that takes a pointer to an integer and multiplies the value by 2. Call the function in 'main()' and print the updated value.

6.4b: Create a function that takes two pointers to integers, swaps their values, and prints the swapped values in 'main()'.

6.4c: Write a function that takes a pointer to an array and its size as arguments, and modifies the array so that all negative numbers are replaced with their absolute values. Demonstrate the function in 'main()'.

6.4d: Implement a function that takes two pointers to integers and returns the result of their addition. Call the function in 'main()' and print the result.

6.4e [Bonus]: Write a C program that uses a function and pointers to compute and print the sum and product of two numbers. Pass the numbers to the function using pointers.

Section A6.5: [More on Functions, Arrays, and Pointers]

6.5a: Implement a C program that reads 5 integers from the user into an array and passes the array to a function that returns the sum of all even numbers using pointers.

6.5b: Create a program that reads 5 integers into an array, passes the array to a function that finds the largest element using pointer arithmetic, and prints the result in 'main()'.

6.5c: Write a function that takes an array, its size, and a pointer to an integer. The function should return the index of the first occurrence of a given integer in the array. If the number is not found, the function should return '-1'. Use the function in 'main()' to search an array.

6.5d: Implement a program where a function takes an array and its size as arguments and prints all the elements that are prime numbers. Use pointers to traverse the array.

6.5e: Write a function that takes an array, its size, and a pointer to the sum and product of the array elements. The function should calculate the sum and product, storing them through pointers. Display the sum and product in 'main()'.