Lab Session 6: Arrays and Pointers

Work on as many exercises as you can during the lab, and finish remaining at home.

Exercise 4 (from lab 5): Pass by reference

You already wrote the C program for the following in Lab 4:

Write a C program that has functions to calculate area of a square, a circle and a rectangle. Ask user for which area is need to be calculated, parameter (length or side of square, radius of circle, etc), and print the resulting area. Use switch to handle the three area conditions.

Now update this code to use pointers. Use void for function returns instead. Pass the parameters by reference to their respective functions. And print their updated value in main using pointers.

Exercise 5 (from lab 5): Palindrome

Write a function in C that checks whether a given string is a palindrome. A phrase is considered a palindrome if, after converting all uppercase letters to lowercase and removing all non-alphanumeric characters, it reads the same forward and backward. Alphanumeric characters include letters and numbers. The function should return true if the given string is a palindrome and false otherwise.

(a) Give the solution without the use of pointers.
(b) Give the solution with the use of pointers.

Example 1:

Input: s = "A man, a plan, a canal: Panama"

Output: true

Explanation: "amanaplanacanalpanama" is a palindrome.

Example 2:

Input: s = "race a car"

Output: false

Explanation: "raceacar" is not a palindrome.

Exercise 1: More on arrays

Write a C program to input elements in array, and

(a) Print the frequency of each element in array.
(b) Print all unique elements.

Sample expected output:

```
Enter size of array: 8
Enter elements in array: 4
6
7
5
6
8
7
5

Frequency of all elements of array :
4 occurs 1 times
6 occurs 2 times
7 occurs 2 times
5 occurs 2 times
8 occurs 1 times

Unique elements in the array are: 4 8
```

Exercise 2: More on 2D arrays

Image is a 2D array and image rotation is a fundamental operation in image processing.

Write a C program that takes an input of n x n 2D array, and rotates the 2D array by

(a) 90 degrees clockwise
(b) 90 degrees anticlockwise

Hint: focus on the boundary elements.

Sample expected output:

```
Enter the size of the array: 3
Enter the elements of the array:
1
2
3
4
5
6
7
8
9
The entered array is:
1 2 3
4 5 6
7 8 9
The rotated array is:
7 4 1
8 5 2
9 6 3
The rotated array is:
3 6 9
2 5 8
1 4 7
```

Exercise 3: String manipulation

Write a C program to input a string from the user. Then remove all repeated characters in a string using loops and print the updated string.

Sample expected output:

```
Enter any string: winter is here
String before removing duplicates: winter is here
String after removing duplicates: winter she
```