Session 8: Arrays

8.1 Introduction

An array is a data structure that is used to implement a sequence of data elements that are of the same data type. An array consists of a group of elements that shared the same name and are accessed by indexing.

In most programming languages each element has the same data type and the array occupies a continuous area of storage – it's a homogenous data structure. The elements in the array are contigous to each other.

8.2 Declaring an Array

Syntax

```
data_type arrayname[constant];
```

Like a varaible an array must be declared using a specific data type and must also have a name. This has an implication on the array elements – that all array elements share:

(i.) array name,

(ii.) data type.

Example 1

```
char  myname[25];
```

8.3 Referencing array elements

Each array element can be accessed through an index, that is relative to the postion the element is in the array. In C arrays are zero indexed, this is to mean that their indices start from zero to n – 1 where n is the number of elements in the array.

Example 2

When assigning a value to an array element we use the following syntax:

```
array_name[index] = value | expression;
```

Example 3

Using an array element to assign a value to a variable uses the following syntax:

```
variable = array_name[index];
```

Exercise

Write a program that will accept the values of an integer array of size 10 and display them.

Assignment

Question 1

Write a program that will accept the values of an integer array of size 10 and then display them in reverse order.

Question 2

Write a program that will accept the values of an integer array of size 10, compute their sum, average and standard deviation, display each of these computations.

Question 3

Write a program to read data items into two float arrays X and Y each of size 20. Compute the products of the corresponding elements in X and Y and store the result in a third array XY, also of size 20. Print a three – column table displaying the arrays X, Y, and XY.