Working with Multiples

Name:

Student ID:

Date:

# Learning Journey and Evidence

## Review array, array index, and loops & arrays.

Array is a special kind of variable, one that stores multiple values instead of a single value.

Array index- The values within an array are called elements. The type of each element is the same as the other elements in the array. Elements of an array are accessed via an index.

The first element has the index 0 while the second has the index 1.

Loops & Arrays-Loops are used to iterate over elements in an array. Common types of loops used with arrays are for loops, while loops, and for-each loops (enhanced for loops).

For Loop- used when the number of iterations is known. Syntax: for (int i = 0; i < array.length; i++) { // access array[i] }

While Loop-Useful when the number of iterations is not predetermined. Syntax: int i = 0; while (i < array.length) { // access array[i]; i++; }

For-Each Loop-Simplifies the syntax for iterating over arrays. Syntax: for (type element : array) { // use element }

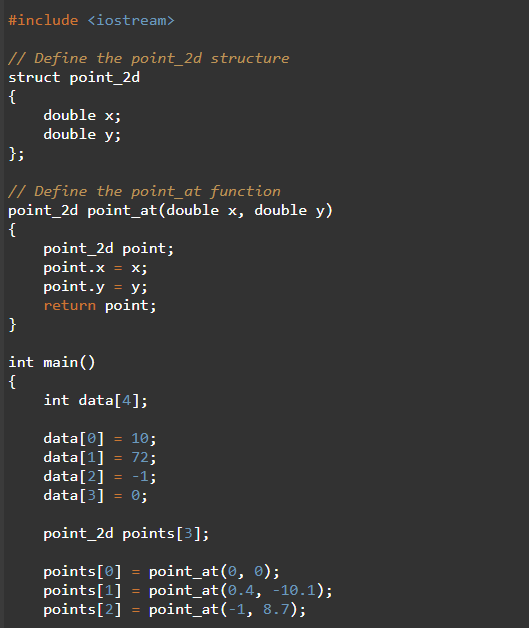
**Resources I used.**

GeeksforGeeks-Comprehensive resource for learning about arrays with numerous examples and practice problems.

Codecademy- Interactive programming courses that cover arrays in various languages.

## Create a small program with array – access each value directly.

Show your process for this and highlight any realisations you gain.



Points Array: We initialize an array of point\_2d with 3 elements and assign coordinates to each point using the point\_at function. We define a structure named point\_2d that contains two double members, x and y, representing the coordinates of the point.

A screen shot of a computer code

Description automatically generated

Printing Points Array: Similarly, we use a for loop to iterate through each element of the points array and print the x and y coordinates of each point.

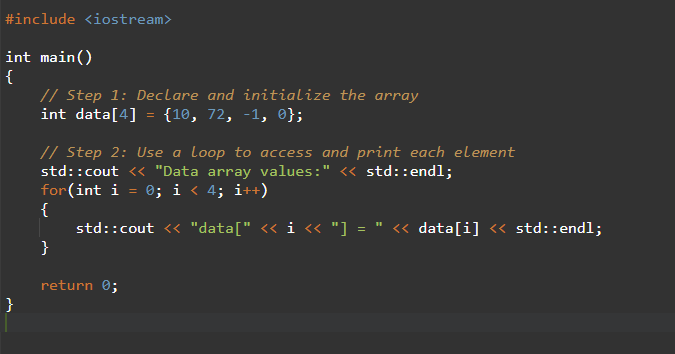
**The OutPut**

**A screenshot of a computer program

Description automatically generated**

## Create a small program with array – use loop to access elements.

Show your process for this and highlight any realisations you gain.

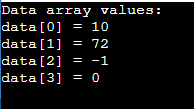


We declare an array of integers named data with 4 elements. We then initialize the array with the values 10, 72, -1, 0

The **for loop** to iterates over the elements of the array.The loop **variable i** starts from 0 and goes up to 3 (since the array has 4 elements).

-Inside the loop, we print each element of the array using its index.

**Output**



**Realization**

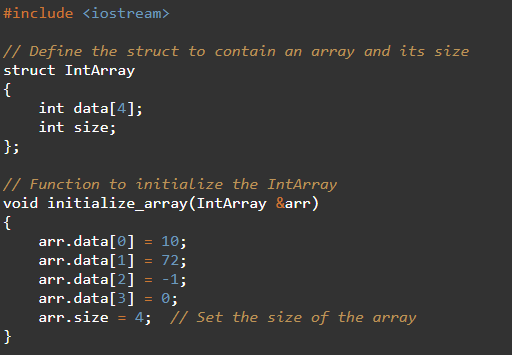
-Using a loop to access array elements is more efficient compared to accessing each element individually.

-The loop ensures that each element is accessed systematically, which is particularly useful for larger arrays where manual access would be impractical.

## Create a small program using struct to contain array and its size.

Show your process for this and highlight any realisations you gain.

Struct Encapsulation- the array and its size in a struct (IntArray), we create a more organized and manageable data structure. This is particularly useful when passing the array and its size around in functions.



Separation of Concerns-The function initialize\_array is responsible for initializing the elements of the array and setting its size. This keeps the main function clean and focused on high-level logic.

Function Calls- In the main function, we declare an IntArray variable and call the initialize\_array function to set it up. We then call the print\_array function to print its contents.

A screen shot of a computer program

Description automatically generated

**Output**

**A black background with a black square

Description automatically generated with medium confidence**

**Realizations**

Encapsulation Benefits-Encapsulating the array and its size in a struct makes the code cleaner and more modular. It also reduces the risk of errors related to mismatched array sizes and indices.

Function Separation-Separating initialization and printing logic into different functions enhances code readability and reusability.

## Build the [Statistics Calculator](https://programmers.guide/book/part-2-organised-code/5-working-with-multiples/2-put-together/01-00-designing-statistics-calculator/) program.

### Create data type, read in values in populate array, output in main.

Show your process for this and highlight any realisations you gain.

### Implement Sum

Show your process for this and highlight any realisations you gain.

### Implement Mean

Show your process for this and highlight any realisations you gain.

### Implement Maximum

Show your process for this and highlight any realisations you gain.

Show hand execution of your maximum function with data [-7, -1, -10], assuming at most 4 values in your array.

### Implement Add Data

Show your process for this and highlight any realisations you gain.

Show hand execution of your add\_data function to add 8 to [0, 5, -2], assuming at most 4 values in your array.

### Implement Remove Data

Show your process for this and highlight any realisations you gain.

Show hand execution of your remove\_data function to remove data at index 1 from [1, 2, 3], assuming at most 4 values in your array.

Show hand execution of your remove\_data function to remove data at index 2 from [1, 2, 3], assuming at most 4 values in your array.

Show hand execution of your remove\_data function to remove data at index -1 from [1, 2, 3], assuming at most 4 values in your array.

## Capture any other study and practice used to master these concepts.

Show evidence of any additional study and practice you did to master these concepts.

## Complete one of the [Test Your Knowledge](https://programmers.guide/book/part-2-organised-code/5-working-with-multiples/4-activities/) activities

Show your process for this and highlight any realisations you gain.

# Brief Summary of Concepts

|  |  |
| --- | --- |
| **Concept** | **Key Idea / Concept** |
| Array | An array is a collection of elements, all the same type, stored in contiguous memory locations. It allows for efficient access and modification of data using indices. |
| Array Index | An array index is an integer value used to access a specific element within an array. Indices typically start at 0 and go up to the array's size minus one. |
| Array and For Loop | A for loop is commonly used to iterate over the elements of an array. It provides a convenient way to access and manipulate each element using its index. |
| Multi-dimensional Array | A multi-dimensional array is an array of arrays. It can be used to represent more complex data structures like matrices. Each element in a multi-dimensional array is accessed using multiple indices. |
| Segmentation Fault | A segmentation fault occurs when a program tries to access a memory location that it is not allowed to access, often due to out-of-bounds array access or dereferencing a null or dangling pointer. |
| Dangling Pointers | A dangling pointer is a pointer that does not point to a valid object or memory location, often because the object it pointed to has been deleted or deallocated. |

# Reflection

## What gives you confidence you have achieved the learning goals?

#### Declare and use arrays within your code in variables, fields, and parameters.

I can confidently declare arrays of different types and sizes and use them in various contexts within my code. For instance, I've created and initialized arrays within functions and structs, as demonstrated in the provided examples.

#### Write code that interacts with the array, and with its elements.

#### I have written multiple pieces of code that access and manipulate array elements directly. This includes initializing arrays, modifying elements, and performing operations on them, such as summing or printing their values.

#### Iterate over the elements in an array to interact with each as an individual value.

Using for loops and other iterative constructs, I can iterate over arrays to process each element individually. This was shown in the print\_array function, where I looped through the array elements and printed each one.

#### Use hand execution to illustrate how code works with an array.

Answer here…

## What is the most important thing you learned from this and why?

The most important thing I learned from this exercise is the value of encapsulating data structures using structs. This approach not only simplifies the management of arrays by keeping related information together but also enhances code readability, maintainability, and safety. By encapsulating the array and its size, I can easily pass this structure to functions and ensure that operations on the array, thereby reducing the risk of errors such as out-of-bounds access.