



UNIVERSITY OF CALOOCAN CITY  
COMPUTER ENGINEERING DEPARTMENT



Data Structure and Algorithm

Laboratory Activity No. 5

---

# Implementation of Arrays

---

*Submitted by:*  
Villanueva, Bryan O.

*Instructor:*  
Engr. Maria Rizette H. Sayo

August 16, 2025

# I. Objectives

## Introduction

Array, in general, refers to an orderly arrangement of data elements. Array is a type of data structure that stores data elements in adjacent locations. Array is considered as linear data structure that stores elements of same data types. Hence, it is also called a linear homogenous data structure.

This laboratory activity aims to implement the principles and techniques in:

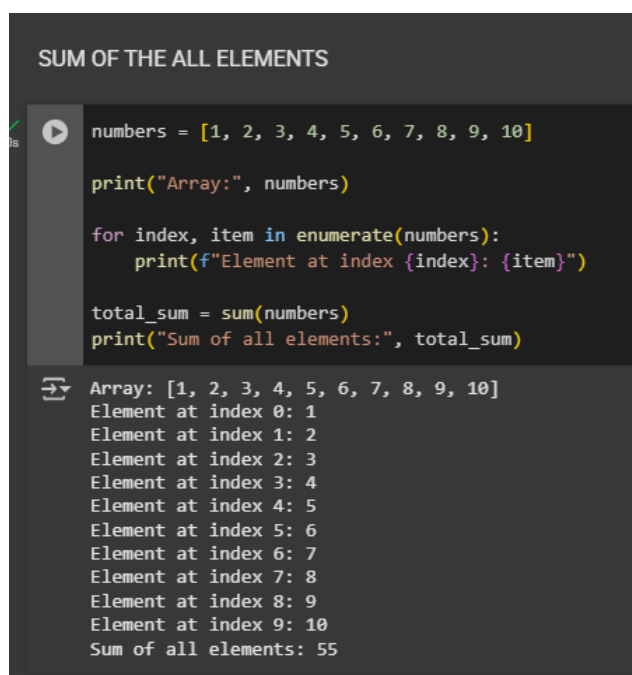
- Writing algorithms using Array data structure
- Writing a python program that can implement Array data structure

# II. Methods

- Write a Python program to create an array of 10 integers and display the array items. Access individual elements through indexes and compute for the sum.
- Write a Python program to append a new item to the end of the array. Original array: numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
- Write a Python program to insert a new item before the second element in an existing array. Original array: numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
- Write a Python program to reverse the order of the items in the array. Original array: numbers = [5, 4, 3, 2, 1]

Write a Python program to get the length of the array. Original array: numbers = [5, 4, 3, 2, 1]

# III. Results



```
SUM OF THE ALL ELEMENTS

numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

print("Array:", numbers)

for index, item in enumerate(numbers):
    print(f"Element at index {index}: {item}")

total_sum = sum(numbers)
print("Sum of all elements:", total_sum)

Array: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
Element at index 0: 1
Element at index 1: 2
Element at index 2: 3
Element at index 3: 4
Element at index 4: 5
Element at index 5: 6
Element at index 6: 7
Element at index 7: 8
Element at index 8: 9
Element at index 9: 10
Sum of all elements: 55
```

Figure 1 Screenshot of program 1

APPEND NEW ITEM

```
[10] numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
    print("Original array:", numbers)

    numbers.append(11)
    print("Appending the new item:", numbers)
```

➤ Original array: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]  
Appending the new item: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]

Figure 2 Screenshot of program 2

INSERTING NEW ITEM

```
[9] numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
    print("Original array:", numbers)

    new_number = int(input("Enter a number to insert before the second element: "))

    numbers.insert(1, new_number)

    print("After inserting", new_number, "before the second element:", numbers)
```

➤ Original array: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]  
Enter a number to insert before the second element: 69  
After inserting 69 before the second element: [1, 69, 2, 3, 4, 5, 6, 7, 8, 9, 10]

Figure 3 Screenshot of program 3

REVERSING THE ARRAY

```
[6] numbers = [5, 4, 3, 2, 1]
    print("Original array:", numbers)

    numbers.reverse()
    print("Reversed array:", numbers)
```

➤ Original array: [5, 4, 3, 2, 1]  
Reversed array: [1, 2, 3, 4, 5]

Figure 4 Screenshot of program 4

GETTING THE LENGTH

```
▶ numbers = [5, 4, 3, 2, 1]
  print("Array:", numbers)

  length = len(numbers)
  print("Length of the array:", length)
```

➤ Array: [5, 4, 3, 2, 1]  
Length of the array: 5

Figure 5 Screenshot of program 5

The program works by creating an array and performing different operations with it. It can display the elements, add a new item at the end, insert an item before the second element, reverse the order of items, and show the length of the array. Each part of the program shows how arrays can be changed and accessed using Python.

## IV. Conclusion

From the results, I realized that arrays are very useful when working with data because they make it easier to organize and access elements. I saw how simple operations like appending, inserting, reversing, and finding the length can change the array in different ways. These results showed me that arrays are flexible and can be used in many situations, especially in problem solving. Doing this activity also helped me understand better how indexing works and how important it is to know the position of each element. Overall, the results proved that arrays are one of the basic but most important tools in programming.

## References

[1] Co Arthur O.. “University of Caloocan City Computer Engineering Department Honor Code,” UCC-CpE Departmental Policies, 2020.

[2] W3schools. (n.d.). Python Arrays. [Www.w3schools.com](https://www.w3schools.com/python/python_arrays.asp).  
[https://www.w3schools.com/python/python\\_arrays.asp](https://www.w3schools.com/python/python_arrays.asp)