UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT

Data Structure and Algorithm

Laboratory Activity No. 5

Implementation of Arrays

Submitted by:
Barbas, Steven Jade P.

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 as 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

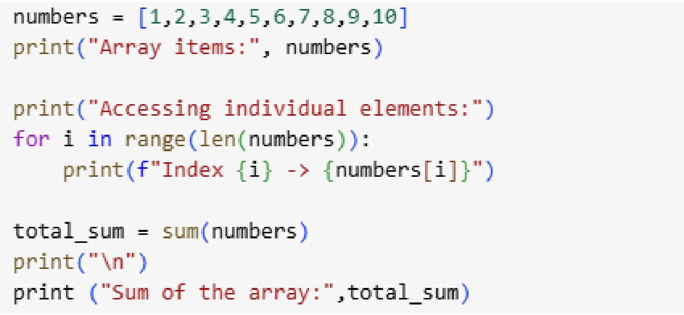
II. Methods

- 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

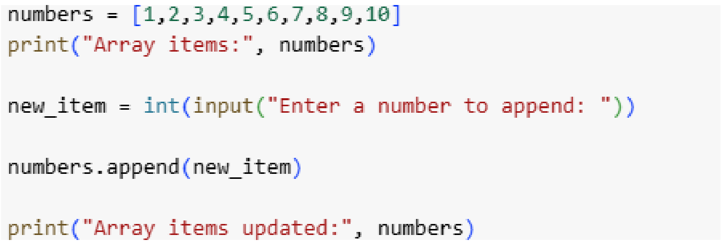
In this laboratory, several Python programs were created to demonstrate implementing Array data structure. An array is a data structure that stores elements of the same type in a sequence. The elements are kept in continuous memory locations, making it a linear structure. Because of this, an array is also called a linear and homogeneous data structure.S



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

print("Accessing individual elements:")
for i in range(len(numbers)):
    print(f"Index {i} -> {numbers[i]}")

total_sum = sum(numbers)
print("\n")
print ("Sum of the array:",total_sum)
```

Figure 1 Screenshot of 1st program

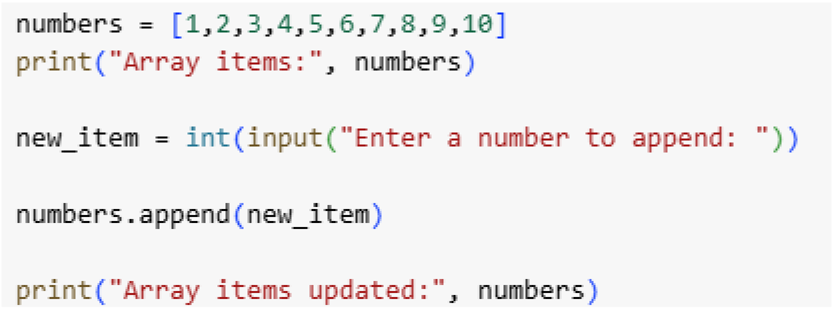
Figure 1 demonstrate a program that creates an array with 10 integers and displays all its elements. It also shows how to access each item using its index and then calculates the total sum of all the numbers in the array.



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

new_item = int(input("Enter a number to append: "))

numbers.append(new_item)

print("Array items updated:", numbers)
```

Figure 2 Screenshot of 2nd program

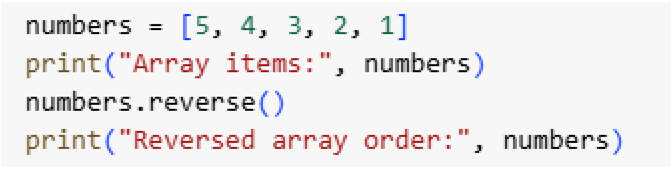
Figure 2 shows program that's adds a new item to the end of the array. This demonstrates how to use the append function in Python to expand an array with additional elements.



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

new_item = int(input("Enter a number to append: "))

numbers.append(new_item)

print("Array items updated:", numbers)
```

Figure 3 Screenshot of 3rd program

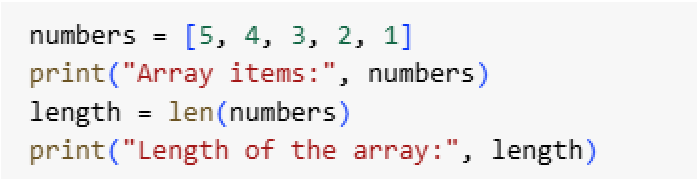
Figure 3 shows program takes the original array and inserts a new item before the second element. This shows how to use array indexing to add elements at a specific position, not just at the end.



```
numbers = [5, 4, 3, 2, 1]
print("Array items:", numbers)
numbers.reverse()
print("Reversed array order:", numbers)
```

Figure 4 Screenshot of 4th program

Figure 4 shows a program that reverses the order of the items in the array. This shows how Python can easily rearrange elements so that the first becomes last and the last becomes first.



```
numbers = [5, 4, 3, 2, 1]
print("Array items:", numbers)
length = len(numbers)
print("Length of the array:", length)
```

Figure 5 Screenshot of 5th program

The figure 5 program finds and displays the length of the array. This shows how to use Python’s built-in function to count the total number of elements in an array.

IV. Conclusion

In conclusion, the activities showed different ways of implementing arrays in Python. We created arrays, accessed elements, computed the sum, added new items, inserted elements at specific positions, reversed the order, and found the length of the array. Through these tasks, we learned the basic operations that make arrays an important data structure for storing and managing data efficiently.

References

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