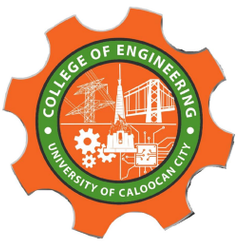
UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT



Data Structure and Algorithm

Laboratory Activity No. 5

Implementation of Arrays

Submitted by:
Sorellano, John Kenneth T.

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

- **ALGORITHM**

1. Start
2. Initialize an array with 10 integers:

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

3. Set total = 0
4 . For each element in the array: Add the element’s value to total
5. After adding all elements, display total
6. End

- **PROGRAM**

```
# array of 10 integers
array = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
total = 0

# summation of array
total += array [0]
total += array [1]
total += array [2]
total += array [3]
total += array [4]
total += array [5]
total += array [6]
total += array [7]
total += array [8]
total += array [9]

print (f"the sum of array:", total)
```

```
the sum of array: 55
```

- **ALGORITHM**

1. Start
2. Initialize an array with 10 integers:

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

3. Display the original array
4. Ask the user to input a new number
5. Store the input number in variable numbers
6. Append numbers to the end of the array
7. Display the updated array
8. End

- **PROGRAM**

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

     #inputing a number
     numbers = int(input("Input a new number:"))

     #appending number
     array.append(numbers)

     print(f"Updated array:{array}")
```

```
Original array: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
Input a new number:1000
Updated array:[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1000]
```

- **ALGORITHM**

1. Start
2. Initialize an array with 10 integers:

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

3. Display the original array

4. Ask the user to input a new number
5. Store the input number in variable numbers
6. Insert numbers into the second position of the array (index = 1)
7. Display the updated array
8. End

- **PROGRAM**

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

     #inputing a number
     numbers = int(input("Input a new number in second element:"))

     #insert a second number in the element
     array.insert(1, numbers)

     print(f"Updated array:{array}")
```

```
Original array: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
Input a new number in second element:1000
Updated array:[1, 1000, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

- **ALGORITHM**

1. Start
2. Initialize an array with values:

   array = [5, 4, 3, 2, 1]
3. Display the original array
4. Reverse the array (so the first element becomes last, and the last becomes first)
5. Display the reversed array
6. End

- **PROGRAM**

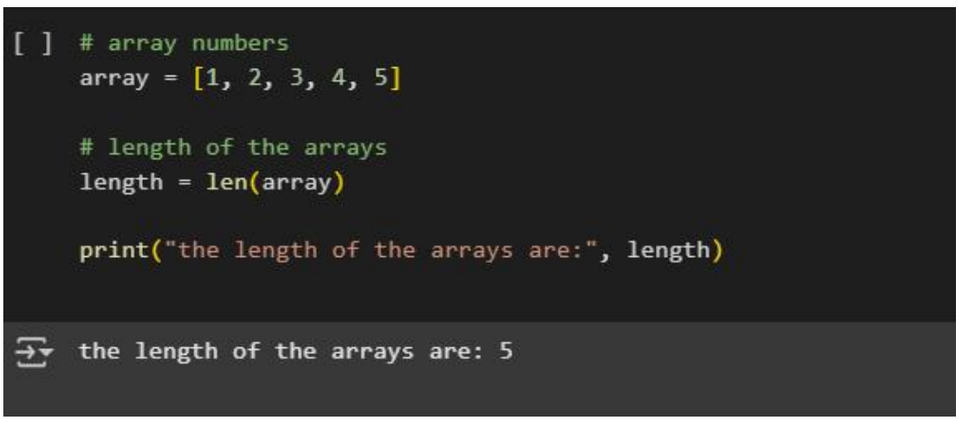
```
#Original array
array = [5, 4, 3, 2, 1]
print(f"Original array:",array)

#Reversing the array
array.reverse()

print(f"Reversed array:",array)
```

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

- **ALGORITHM**

1. Start
2. Initialize an array with values:
   array = [1, 2, 3, 4, 5]
3. Find the length of the array using len(array)
4. Store the result in variable length
5. Display length
6. End

- **PROGRAM**



IV. Conclusion

In conclusion, all of the programs that I made helps me to understand how array works. This programs shows basic array operations such as summing elements, appending a new number, inserting at a specific position, reversing the order, and finding the array's length.

References

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