



UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT



Data Structure and Algorithm

Laboratory Activity No. 5

Implementation of Arrays

Submitted by:
Dispo, Lei Andrew T.

Instructor:
Engr. Maria Rizette H. Sayo

08, 23, 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. Create an array with elements : [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
3. Initialize a variable total = 0
4. For each variable in the array add element's value in the total
5. Display "The sum of array:" and the value of total
6. End

```

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

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. Create an array with elements: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
3. Display “Original array:” and add array
4. Ask the use to **Input new number in the array**
5. Append numbers to the array
6. Display “New array:” and the updated array
7. End

```

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

numbers=int(input("Input new number:"))

array.append(numbers)

print (f"new array:", array)

```

- **ALGORITHM**

1. Start
2. Create an array with elements: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
3. Display “Original array:” and the array
4. Ask the user to **Input new number** and store it in numbers
5. Insert numbers into the array at position index 1
6. Display “New array:” and the updated array
7. End

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

numbers = int(input("Input new number:"))

array.insert(1, numbers)

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

- **ALGORITHM**

1. Start
2. Create an array with elements : [5, 4, 3, 2, 1]
3. Display “Original array:” and the array
4. Reverse the array elements (so the first becomes last, and last becomes first)
5. Display “New array:” and the reversed array
6. End

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

array.reverse()
|
print(f"New array:", array)
```

Original array: [5, 4, 3, 2, 1]
New array: [1, 2, 3, 4, 5]

- **ALGORITHM**

1. Start
2. Create an array with elements [1, 2, 3, 4, 5]
3. Find the number of elements in the array using len(array)
4. Display “Length of array:” and the value of length

5. End

```
array = [1, 2, 3, 4, 5]

length = len(array)

print(f"Length of array:", length)
```

· Length of array: 5

IV. Conclusion

- I learned in this laboratory how to use array and adding length or appending.

References

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