



UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT



Data Structure and Algorithm

Laboratory Activity No. 5

Implementation of Arrays

Submitted by:
Acebedo, Sebastian C.

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:

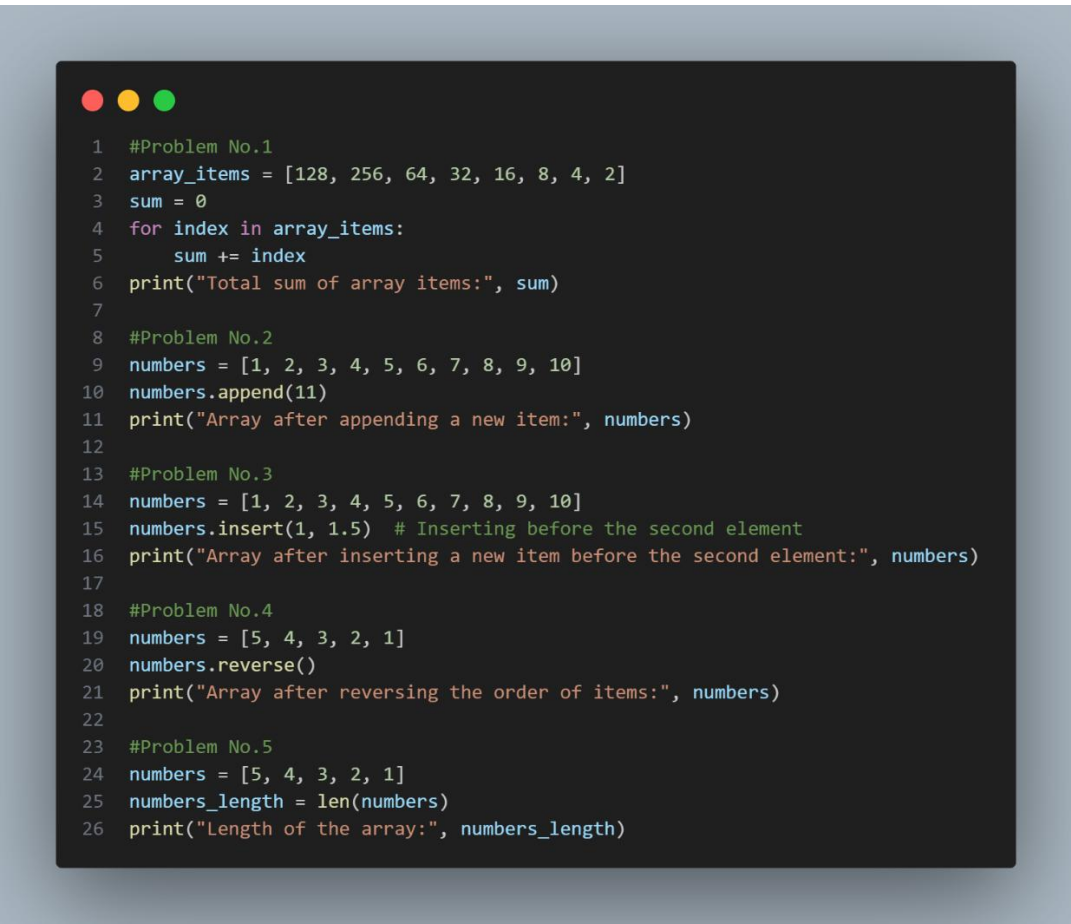
- Writing algorithms using Array data structure
- Writing a python program that can implement Array data structure

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

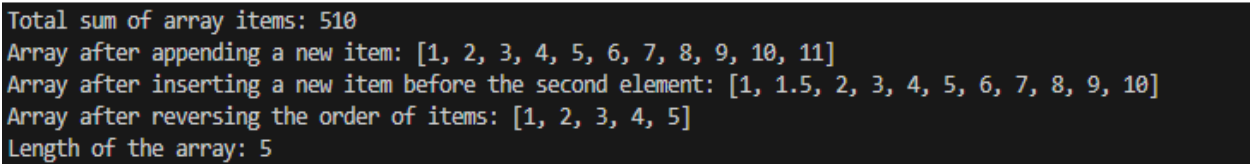


```
1  #Problem No.1
2  array_items = [128, 256, 64, 32, 16, 8, 4, 2]
3  sum = 0
4  for index in array_items:
5      sum += index
6  print("Total sum of array items:", sum)
7
8  #Problem No.2
9  numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
10 numbers.append(11)
11 print("Array after appending a new item:", numbers)
12
13 #Problem No.3
14 numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
15 numbers.insert(1, 1.5) # Inserting before the second element
16 print("Array after inserting a new item before the second element:", numbers)
17
18 #Problem No.4
19 numbers = [5, 4, 3, 2, 1]
20 numbers.reverse()
21 print("Array after reversing the order of items:", numbers)
22
23 #Problem No.5
24 numbers = [5, 4, 3, 2, 1]
25 numbers_length = len(numbers)
26 print("Length of the array:", numbers_length)
```

Figure 1 Screenshot of program

This code represents different data arrays and demonstrates fundamental operations on them. Each section showcases a specific array manipulation technique:

- The first part illustrates calculating the total sum of an array’s elements.
- The second part shows how to append a new item to the end of an array.
- The third part demonstrates inserting a new item before a specific position in the array.
- The fourth part reverses the order of items in an array.
- The fifth part calculates and displays the length of an array.



```
Total sum of array items: 510
Array after appending a new item: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]
Array after inserting a new item before the second element: [1, 1.5, 2, 3, 4, 5, 6, 7, 8, 9, 10]
Array after reversing the order of items: [1, 2, 3, 4, 5]
Length of the array: 5
```

Figure 2 Screenshot of Program

This image represents the output of the code. It shows the results of different array operations, like calculating the total sum, adding a new item, inserting an item at a specific position, reversing the array, and finding the length of the array. Each output helps to understand how arrays can be changed and what happens after each operation.

Please refer to this link:



DSA-Lab-Report-5.py

IV. Conclusion

In conclusion, this code demonstrates several basic array operations in Python, such as summing elements, appending and inserting items, reversing the order, and finding the length of an array. Each part shows how arrays can be changed and analyzed using simple commands. By practicing these operations, I learned how to manage and work with arrays more effectively in my programs.

References

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