Data Structure and Algorithm

Laboratory Activity No. 4

Arrays

|  |  |
| --- | --- |
| *Submitted by:* | *Instructor:* |
| Caasi, Karl Benedict D. | Engr. Maria Rizette H. Sayo |

August 09, 2025

# 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
* Solve programming problems using dynamic memory allocation, arrays and pointers

# Methods

Jenna’s Grocery

A list of grocery items

AI-generated content may be incorrect.

Jenna wants to buy the following fruits and vegetables for her daily consumption. However, she needs to distinguish between fruit and vegetable, as well as calculate the sum of prices that she has to pay in total.

Problem 1: Create a class for the fruit and the vegetable classes. Each class must have a constructor, deconstructor, copy constructor and copy assignment operator. They must also have all relevant attributes (such as name, price and quantity) and functions (such as calculate sum) as presented in the problem description above.

Problem 2: Create an array GroceryList in the driver code that will contain all items in Jenna’s Grocery List. You must then access each saved instance and display all details about the items.

Problem 3: Create a function TotalSum that will calculate the sum of all objects listed in Jenna’s Grocery List.

Problem 4: Delete the Lettuce from Jenna’s GroceryList list and de-allocate the memory assigned.

A screen shot of a computer program

AI-generated content may be incorrect.

Problem 1

Figure 1 Screenshot of program

You define a base class Item with attributes: name, price, and qty.Both shallow and deep copy create new Apple objects with the same attributes. The print statements inside the copy methods help you track when shallow and deep copies happen.

The total price calculation and display confirm that copies have the same data as the original.

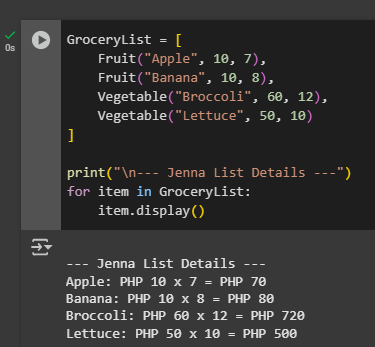
Problem 2

Figure 2

You create a list called GroceryList containing multiple Fruit and Vegetable objects with different names, prices, and quantities.You successfully created a grocery list with fruits and vegetables. Constructor messages confirm creation of each Item. Display shows price calculations for each item.

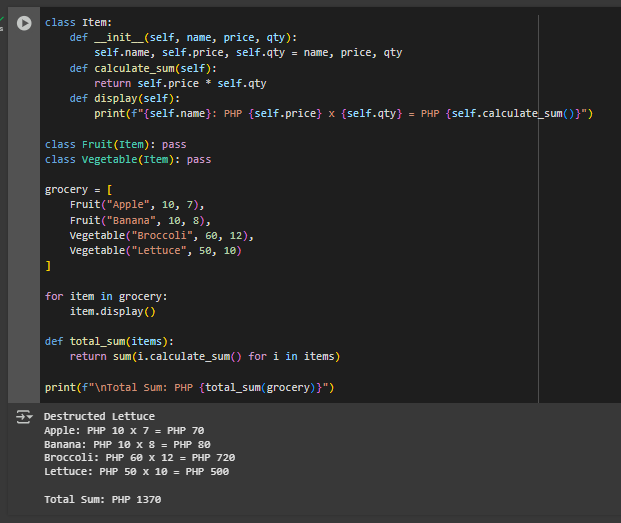
Problem 3

Figure 3

The class has an \_\_in it\_\_ method that initializes the name, price, and quantity; a calculate sum() method that returns the total cost by multiplying price and quantity; and a display () method that prints the item details in the format: Name: PHP price x qty = PHP total.

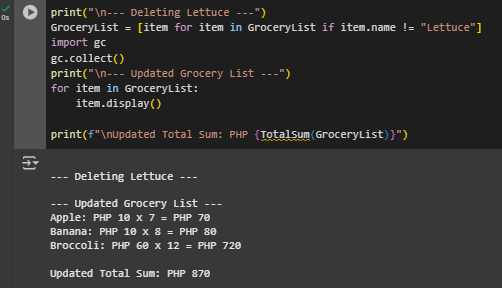
Problem 4

Figure 4

The code removes the item "Lettuce" from the grocery list using a list comprehension, then calls the garbage collector to clean up, and finally prints the updated list and total cost without Lettuce.

**Conclusion**This project successfully demonstrates the concepts of object-oriented programming by creating classes for grocery items and their subclasses. It shows how constructors and destructors work, how to implement shallow and deep copies, and how to manage a list of objects effectively. The program calculates individual item costs and totals, updates the grocery list dynamically by removing items, and reflects changes correctly in the output. Overall, this exercise highlights fundamental Python programming techniques such as inheritance, copying, list comprehension, and memory management, which are essential skills for building efficient and maintainable code.  
  
**Reference**<https://docs.python.org/3/tutorial/classes.html>  
https://realpython.com/python-lists-tuples/  
https://docs.python.org/3/library/copy.html