#### 1. array\_problem1.py: Traversing an Array

def traverse\_array(arr):

for elem in arr:

print(elem, end=" ")

print()

if \_\_name\_\_ == "\_\_main\_\_":

arr = [1, 2, 3, 4, 5]

print("Array elements:", end=" ")

traverse\_array(arr)

#### array\_problem2.py: Searching for an Element in an Array

def linear\_search(arr, target):

for index, value in enumerate(arr):

if value == target:

return index

return -1 # Return -1 if element is not found

if \_\_name\_\_ == "\_\_main\_\_":

arr = [10, 20, 30, 40, 50]

target = 30

result = linear\_search(arr, target)

if result != -1:

print(f"Element {target} found at index {result}")

else:

print(f"Element {target} not found!")

#### array\_problem3.py: Sorting an Array (Bubble Sort)

def bubble\_sort(arr):

n = len(arr)

for i in range(n):

for j in range(0, n - i - 1):

if arr[j] > arr[j + 1]:

arr[j], arr[j + 1] = arr[j + 1], arr[j]

def print\_array(arr):

for elem in arr:

print(elem, end=" ")

print()

if \_\_name\_\_ == "\_\_main\_\_":

arr = [5, 2, 9, 1, 5, 6]

print("Original array:", end=" ")

print\_array(arr)

bubble\_sort(arr)

print("Sorted array:", end=" ")

print\_array(arr)

#### Summary of Structure:

1. **Traversal**: A simple iteration through the array to print all elements.
2. **Search**: A linear search that finds the index of a target element in the array.
3. **Sorting**: The implementation of **Bubble Sort**, a basic sorting algorithm.
4. **Reversing**: A function to reverse the array by swapping the elements.
5. **Finding Largest and Smallest**: A function that returns the largest and smallest elements in an array.

#### Tips for Organizing the Repository:

1. **Naming conventions**: Use meaningful names for each file, such as array\_problem1.py, array\_problem2.py, etc., so it's easy to track which problem is solved in which file.
2. **Comments**: Add comments to each function to explain its purpose and logic, especially for complex algorithms or steps that may need explanation later.
3. **Testing**: Make sure to test each program with different test cases to ensure they work as expected before committing them.