**TASK 5: Implement various Searching and Sorting Operations in python programming**

**(CO2-K3)**

**Task 5 (a) Binary Search**

Given an array of integers nums which is sorted in ascending order, and an integer target, write a function to search target in nums. If target exists, then return its index. Otherwise, return -1. Apply an algorithm with O(log n) runtime complexity.

**Example 1:**

**Input:** nums = [-1,0,3,5,9,12], target = 9

**Output:** 4

**Explanation:** 9 exists in nums and its index is 4

**Example 2:**

**Input:** nums = [-1,0,3,5,9,12], target = 2

**Output:** -1

**Explanation:** 2 does not exist in nums so return -1

**Constraints:**

* 1 <= nums.length <= 104
* -104 < nums[i], target < 104
* All the integers in nums are **unique**.
* nums is sorted in ascending order.

**Program:**

def binary\_search(nums, target):

low = 0

high = len(nums) - 1

while low <= high:

mid = (low + high) // 2 # middle index

if nums[mid] == target:

return mid

elif nums[mid] < target:

low = mid + 1

else:

high = mid - 1

return -1

nums = [-1, 0, 3, 5, 9, 12]

target = 9

print("Input:", nums, "Target:", target)

print("Output:", binary\_search(nums, target))

target = 2

print("Input:", nums, "Target:", target)

print("Output:", binary\_search(nums, target))

**Task 5 (b) Find the sequence of given integers**

Sort the given set of integers using bubble sort and find the smallest and largest among given set of integers.

Step 1: Get the number of integers

Step 2: Receive the integers

Step 3: Sort the integers using bubble sort

Step 4: Print the start and end of sequence

**Test case 1**

**INPUT**

4

99

5

6

97

**OUTPUT**

Sequence of integers: 4 to 100

**Program:**

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] # swap

return arr

n = int(input("Enter number of integers: "))

arr = []

for i in range(n):

num = int(input(f"Enter integer {i+1}: "))

arr.append(num)

sorted\_arr = bubble\_sort(arr)

smallest = sorted\_arr[0] - 1

largest = sorted\_arr[-1] + 1

print("Sorted integers:", sorted\_arr)

print("Sequence of integers:", smallest ,largest)

**Problem 3:**

A library stores the ID numbers of books in a list. Write a Python program to **search for a particular book ID** entered by the user using **Linear Search**. If the book is available, display its position in the list; otherwise, display "Book not found."

**Program:**

def linear\_search(book\_list, key):

for i in range(len(book\_list)):

if book\_list[i] == key:

return i

return -1

books = [101, 205, 150, 320, 450, 600]

book\_id = int(input("Enter Book ID to search: "))

pos = linear\_search(books, book\_id)

if pos != -1:

print("Book found at position", pos)

else:

print("Book not found")