#include <iostream>

#include <vector>

#include <algorithm> // For sort

#include <cstdlib>   // For rand and srand

#include <ctime>     // For time

// Function to perform binary search

int binarySearch(const std::vector<int>& arr, int left, int right, int key) {

    if (left <= right) {

        int mid = left + (right - left) / 2;

        // Check if the key is at mid

        if (arr[mid] == key)

            return mid;

        // If key is smaller, search in the left half

        if (arr[mid] > key)

            return binarySearch(arr, left, mid - 1, key);

        // Else search in the right half

        return binarySearch(arr, mid + 1, right, key);

    }

    // Return -1 if the element is not present

    return -1;

}

int main() {

    std::vector<int> arr;

    int n = 5000; // Size of the array

    int key;

    // Initialize random seed

    srand(static\_cast<unsigned>(time(0)));

    // Fill the array with random integers

    for (int i = 0; i < n; i++) {

        arr.push\_back(rand() % 10000); // Random number between 0 and 9999

    }

    // Sort the array

    std::sort(arr.begin(), arr.end());

    // Display the array

    std::cout << "Array: ";

    for (int i = 0; i < n; i++) {

        std::cout << arr[i] << " ";

    }

    std::cout << std::endl;

    // Get the key element from the user

    std::cout << "\nEnter the key to search: ";

    std::cin >> key;

    // Perform binary search

    int result = binarySearch(arr, 0, arr.size() - 1, key);

    // Output the result

    if (result != -1) {

        std::cout << "Element found at index " << result << std::endl;

    } else {

        std::cout << "Element not found in the array" << std::endl;

    }

    return 0;

}