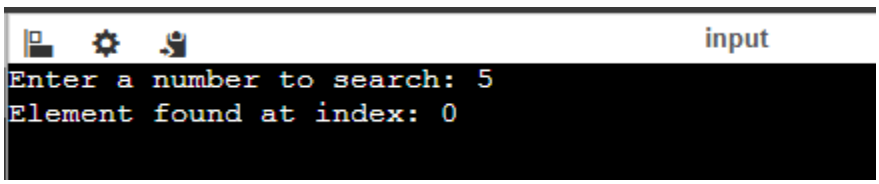
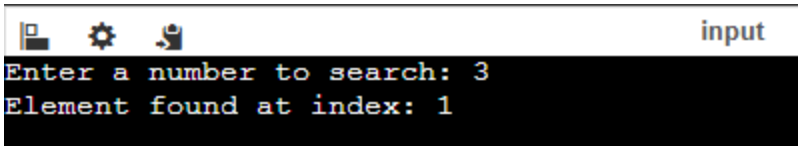
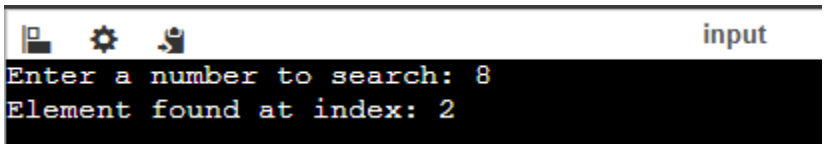
EXERCISE 6:

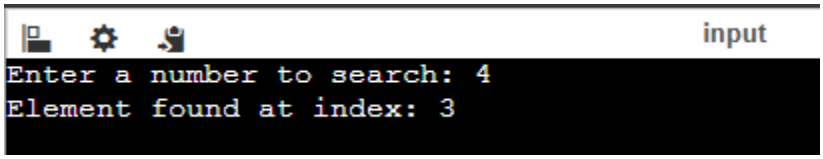
LINEAR SEARCH:

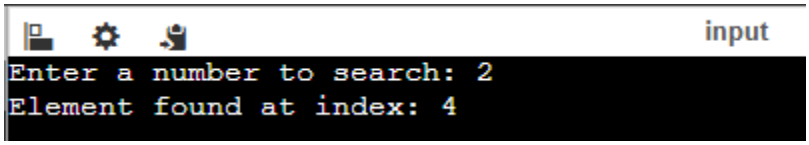
```
C/C++
#include <iostream>
using namespace std;

int linearSearch(int arr[], int size, int target) {
    for (int i = 0; i < size; i++) {
        if (arr[i] == target) {
            return i; // Return the index if found
        }
    }
    return -1; // Return -1 if not found, if may error nothing dun sa array is
target
}

int main() {
    int arr[] = {5, 3, 8, 4, 2};
    int size = sizeof(arr) / sizeof(arr[0]);
    int target;

    cout << "Enter a number to search: ";
    cin >> target;

    int result = linearSearch(arr, size, target);
    if (result != -1) {
        cout << "Element found at index: " << result << endl;
    } else {
        cout << "Element not found." << endl;
    }

    return 0;
}
```

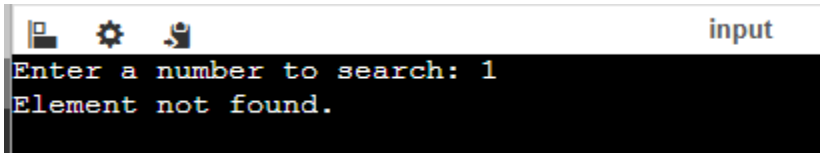












BINARY SEARCH:

C/C++

```
#include <iostream>
using namespace std;

int binarySearch(int arr[], int size, int target) {
  int left = 0;
  int right = size - 1;

  while (left <= right) {
    int mid = left + (right - left) / 2;
    if (arr[mid] == target) {
      return mid; // Return the index if found
    }
    if (arr[mid] < target) {
      left = mid + 1; // Search in the right half
    } else {
      right = mid - 1; // Search in the left half
    }
  }
  return -1; // Return -1 if not found
```

```
}

int main() {
  int arr[] = {2, 3, 4, 5, 8}; // This array must be sorted
  int size = sizeof(arr) / sizeof(arr[0]);
  int target;

  cout << "Enter a number to search: ";
  cin >> target;

  int result = binarySearch(arr, size, target);
  if (result != -1) {
    cout << "Element found at index: " << result << endl;
  } else {
    cout << "Element not found." << endl;
  }

  return 0;
}
```