



Data Structures - Lab

Sir. Humayun

Practical #2 - Lab Task #2

Group Members

Abdullah Imran (398)
Laiba Butt (266)
Husnain Raza (282)

Submission Date

29th September, 2025 (Monday)

Lab Task #2

Problem Statement:

A university wants to create a simple search system for its **student ID records**. The system should allow searching for a student ID using both **Linear Search** and **Binary Search** methods.

Instructions:

1. Create an array of **10 student IDs** (e.g., {101, 105, 110, 120, 125, 130, 140, 150, 160, 170}).
2. Implement **Linear Search** to search for a student ID entered by the user.
3. Implement **Binary Search** on the **sorted array** for the same user input.
4. Display the position of the student ID if found, otherwise display "Not Found".
5. Compare the number of comparisons made by both algorithms.

Expected Output

```
Enter Student ID to search: 120
➡ Linear Search → Found at index 3 (Comparisons: 4)
      Binary Search → Found at index 3 (Comparison s: 2)
```

Code

```
#include <iostream>

using namespace std;

int linearSearch(int arr[], int size, int target, int& comparisons) {
    comparisons = 0;
    for (int i = 0; i < size; i++) {
        comparisons++;
        if (arr[i] == target) {
            return i;
        }
    }
    return -1;
}

int binarySearch(int arr[], int size, int target, int& comparisons) {
    comparisons = 0;
    int left = 0;
```

```

int right = size - 1;
int mid;

while (left <= right) {
    comparisons++;
    mid = (left + right) / 2;

    if (arr[mid] == target) {
        return mid;
    }
    else if (arr[mid] < target) {
        left = mid + 1;
    }
    else {
        right = mid - 1;
    }
}
return -1;
}

void displayArray(int arr[], int size) {
    cout << "Student IDs: ";
    for (int i = 0; i < size; i++) {
        cout << arr[i];
        if (i < size - 1) cout << ", ";
    }
    cout << endl;
}

int main() {
    int studentIDs[10] = {101, 105, 110, 120, 125, 130, 140, 150, 160, 170};
    int size = sizeof(studentIDs) / sizeof(int);
    int searchID;
    int linearComparisons, binaryComparisons;

    displayArray(studentIDs, size);
    cout << endl;

    cout << "Enter Student ID to search: ";
    cin >> searchID;

    int linearResult = linearSearch(studentIDs, size, searchID, linearComparisons);

    int binaryResult = binarySearch(studentIDs, size, searchID, binaryComparisons);

    if (linearResult != -1) {
        cout << "Linear Search → Found at index " << linearResult << " (Comparisons: " << linearCompariso
ns << ")" << endl;
}

```

```

} else {
    cout << "Linear Search → Not Found (Comparisons: " << linearComparisons << ")" << endl;
}

if (binaryResult != -1) {
    cout << "Binary Search → Found at index " << binaryResult << " (Comparisons: " << binaryComparisons << ")" << endl;
} else {
    cout << "Binary Search → Not Found (Comparisons: " << binaryComparisons << ")" << endl;
}

return 0;
}

```

Output

Between Index

```

~/University/Data Structures/practicals (4.407s)

./lab-task-2

Student IDs: 101, 105, 110, 120, 125, 130, 140, 150, 160, 170

Enter Student ID to search: 150
Linear Search → Found at index 7 (Comparisons: 8)
Binary Search → Found at index 7 (Comparisons: 2)

```

Last Index

```

~/University/Data Structures/practicals (6.2s)

./lab-task-2

Student IDs: 101, 105, 110, 120, 125, 130, 140, 150, 160, 170

Enter Student ID to search: 170
Linear Search → Found at index 9 (Comparisons: 10)
Binary Search → Found at index 9 (Comparisons: 4)

```

Not Found

~/University/Data Structures/practicals (4.536s)

./lab-task-2

Student IDs: 101, 105, 110, 120, 125, 130, 140, 150, 160, 170

Enter Student ID to search: 200

Linear Search → Not Found (Comparisons: 10)

Binary Search → Not Found (Comparisons: 4)