Foundation University

School of Science and Technology

Data Structure

Lab Manual 04

Submitted by: Warda Javed

094

Submitted to: Sir Sharjeel

Department of Engineering Technology

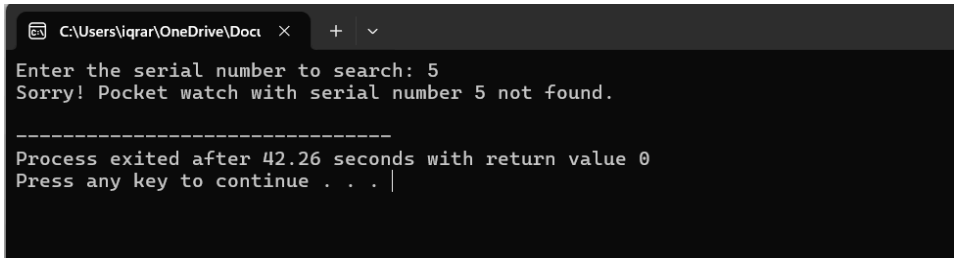
LAB TASK

Searching and Sorting

Question no.1

Code:

```
#include <iostream>
using namespace std;
int main() {
    int serialNumbers[] = {1023, 1045, 1101, 1200, 1309, 1456, 1500};
    int n = sizeof(serialNumbers) / sizeof(serialNumbers[0]);
    int target;
    cout << "Enter the serial number to search: ";
    cin >> target;
    bool found = false;
    // Sequential (Linear) Search
    for (int i = 0; i < n; i++) {
        if (serialNumbers[i] == target) {
            cout << "Vintage pocket watch found at position " << i + 1 << endl;
            found = true;
            break;
        }
    }
    if (!found)
        cout << "Sorry! Pocket watch with serial number " << target << " not found." << endl;
    return 0;
}
```

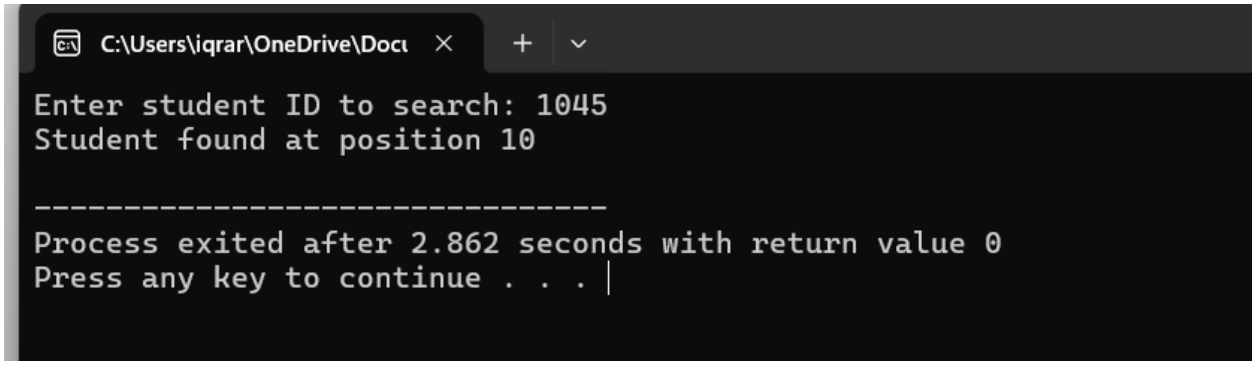


Question no.2

Code:

```
#include <iostream>
using namespace std;
int main() {
    int studentIDs[10] = {1001, 1005, 1010, 1015, 1020, 1025, 1030, 1035, 1040, 1045};
    int target;
    cout << "Enter student ID to search: ";
    cin >> target;
    int low = 0, high = 9;
    bool found = false
    // Binary Search works only on sorted data
    while (low <= high) {
        int mid = (low + high) / 2;
        if (studentIDs[mid] == target) {
            cout << "Student found at position " << mid + 1 << endl;
            found = true;
            break;
        }
        else if (studentIDs[mid] < target)
            low = mid + 1; // Search in right half
        else
            high = mid - 1; // Search in left half
    }
    if (!found)
```

```
    cout << "Student with ID " << target << " not found." << endl;
  return 0;
}
```



Question no.3

Code:

```
#include <iostream>
using namespace std;
int main() {
  int scores[15] = {78, 56, 89, 45, 67, 90, 32, 71, 80, 55, 99, 62, 50, 76, 88};
  int n = 15;
  // Bubble Sort Algorithm
  for (int i = 0; i < n - 1; i++) {
    for (int j = 0; j < n - i - 1; j++) {
      if (scores[j] > scores[j + 1]) {
        // swap scores
        int temp = scores[j];
        scores[j] = scores[j + 1];
        scores[j + 1] = temp;
      }
    }
  }
  cout << "Scores in Ascending Order:\n";
  for (int i = 0; i < n; i++)
```

```
        cout << scores[i] << " ";
    cout << endl;
    return 0;
}
```