DSA LAB 13:

Name: Saif Majid Khan

SAP-ID: 57114

11/17/2024

GitHub:

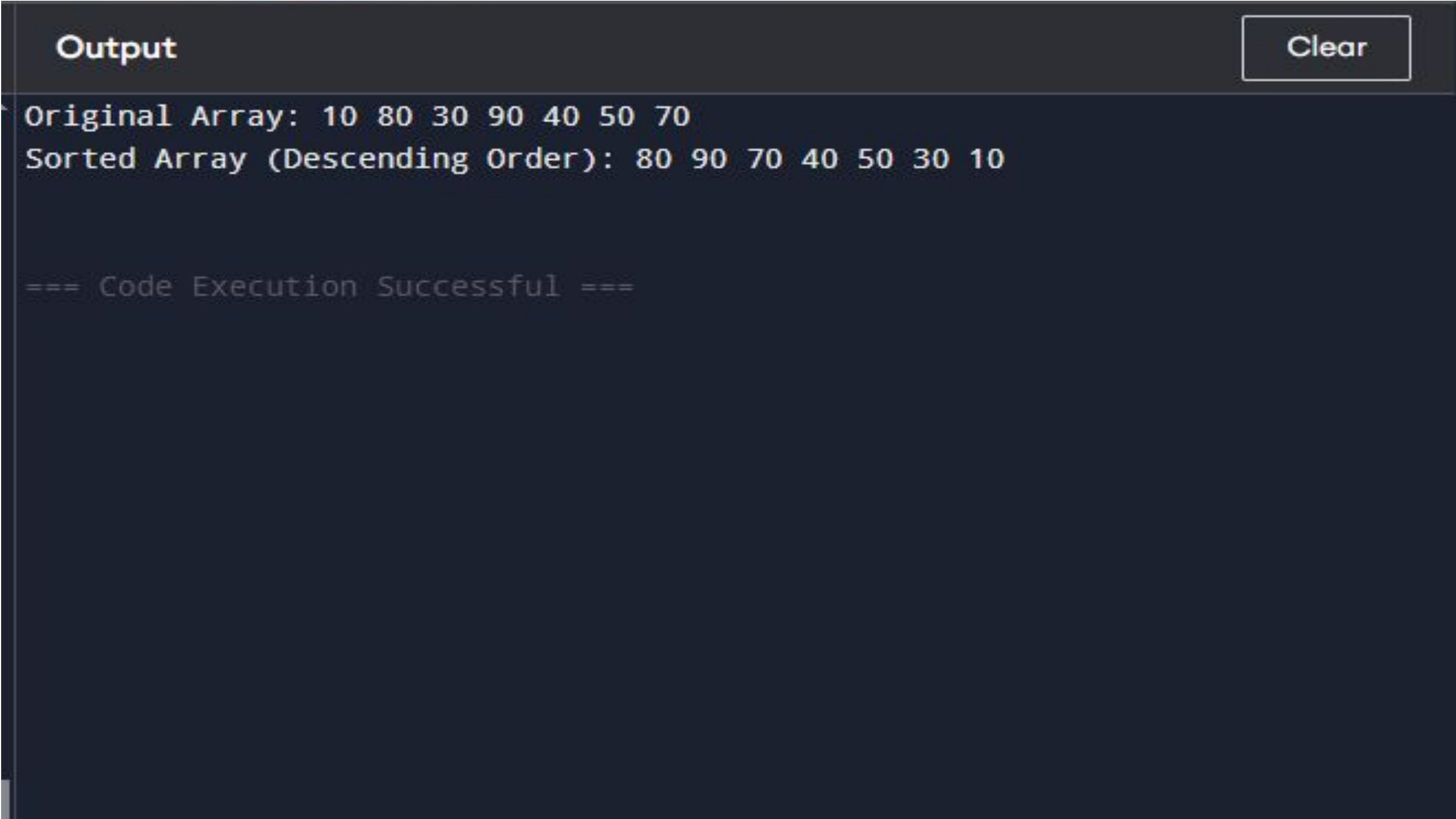
https://github.com/saif01234567/Lab-Tasks-DS

TASK#1:

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

// Function to partition the array for descending order
int partition(int arr[], int first, int last) {
    int pivot = arr[first];
    int bottom = first + 1, top = last;
    int temp;
while (true) {

        while (arr[top] < pivot && top > bottom) {
            top--;
        }
    while (arr[bottom] > pivot && bottom < top) {
        bottom++;
    }

    // If `bottom` and `top` cross, exit loop
    if (bottom >= top) {
        break;
    } else {
        // Swap elements at `bottom` and `top`
        temp = arr[bottom];
        arr[bottom] = arr[top];
        arr[top] = temp;
    }
}
```

```
    temp = arr[first];
    arr[first] = arr[top];
    arr[top] = temp;

    return top; // Return the partition index
}

// Quick Sort function for descending order
void quickSort(int arr[], int first, int last) {
    if (first < last) {
        int pivotIndex = partition(arr, first, last);
        quickSort(arr, first, pivotIndex - 1); // Sort left partition
        quickSort(arr, pivotIndex + 1, last); // Sort right partition
    }
}
```

```
int main() {
    const int size = 7;
    int arr[size] = {10, 80, 30, 90, 40, 50, 70};

    cout << "Original Array: ";
    for (int i = 0; i < size; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;

    quickSort(arr, 0, size - 1);

    cout << "Sorted Array (Descending Order): ";
    for (int i = 0; i < size; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;
    return 0;
}
```



```
Original Array: 10 80 30 90 40 50 70
Sorted Array (Descending Order): 80 90 70 40 50 30 10



=== Code Execution Successful ===
```

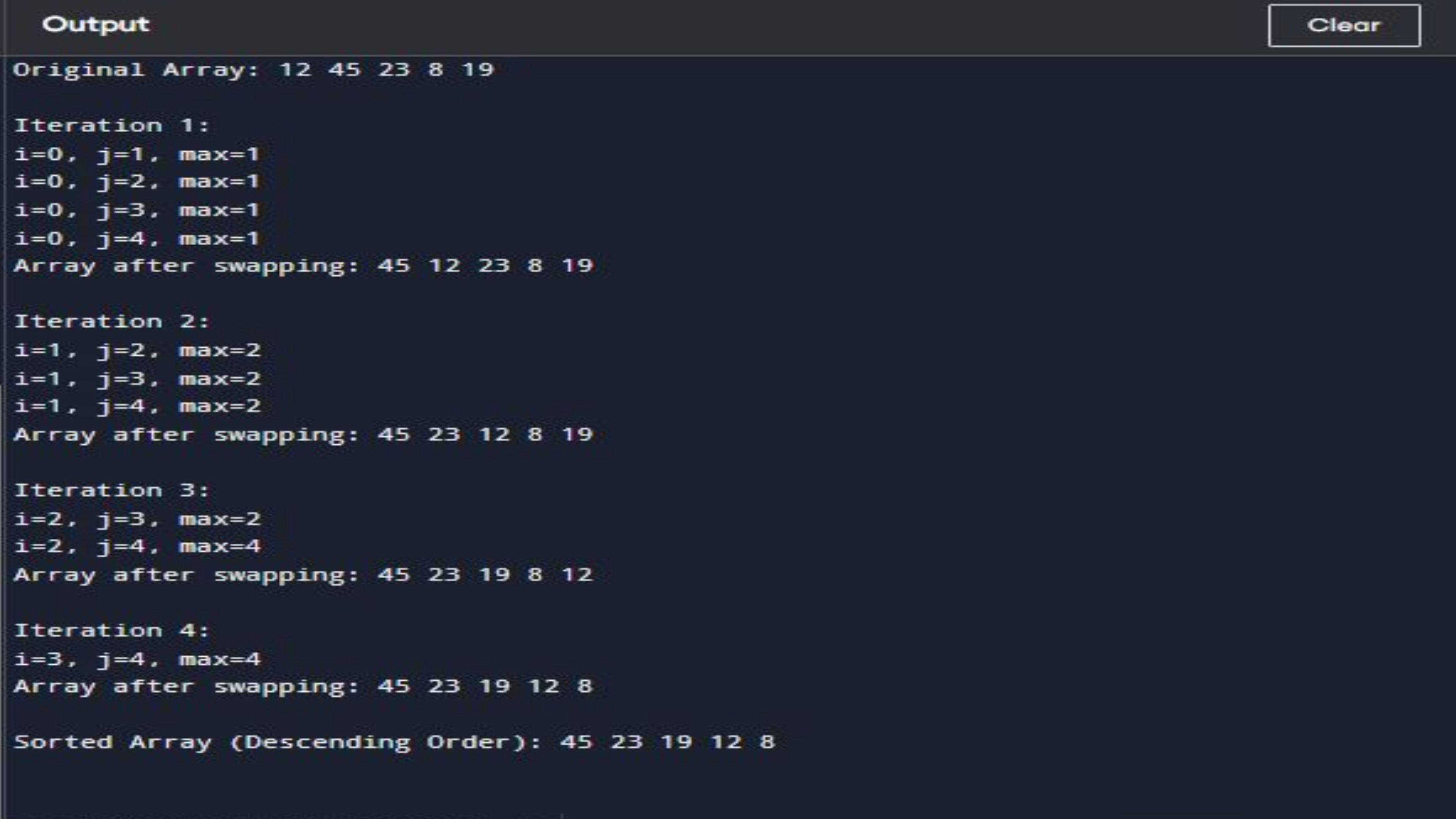
TASK#2

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

// Function to perform Selection Sort in descending order
void SelectionSortDescending(int arr[], int n) {
    int i, j, max, temp;
    for (i = 0; i < n - 1; i++) {
        max = i;
        cout << "\nIteration " << i + 1 << ":" << endl;
```

```
for (j = i + 1; j < n; j++) {
      if (arr[j] > arr[max]) {
        max = j;
      }
      // Display variable values
      cout << "i=" << i << ", j=" << j << ", max=" << max << endl;
    }

    // Swap the elements
    temp = arr[max];
    arr[max] = arr[i];
    arr[i] = temp;
```

```
        // Display the array after the current iteration
        cout << "Array after swapping: ";
        for (int k = 0; k < n; k++) {
            cout << arr[k] << " ";
        }
        cout << endl;
    }
}
int main() {
    const int size = 5; // Fixed array size
    int arr[size] = {12, 45, 23, 8, 19}; // Example array

    cout << "Original Array: ";
    for (int i = 0; i < size; i++) {
        cout << arr[i] << " ";
    }
```

```
 cout << endl;

    // Sort the array in descending order
    SelectionSortDescending(arr, size);

    cout << "\nSorted Array (Descending Order): ";
    for (int i = 0; i < size; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;

    return 0;
}
```



```
Original Array: 12 45 23 8 19

Iteration 1:
i=0, j=1, max=1
i=0, j=2, max=1
i=0, j=3, max=1
i=0, j=4, max=1
Array after swapping: 45 12 23 8 19

Iteration 2:
i=1, j=2, max=2
i=1, j=3, max=2
i=1, j=4, max=2
Array after swapping: 45 23 12 8 19

Iteration 3:
i=2, j=3, max=2
i=2, j=4, max=4
Array after swapping: 45 23 19 8 12

Iteration 4:
i=3, j=4, max=4
Array after swapping: 45 23 19 12 8

Sorted Array (Descending Order): 45 23 19 12 8
```