ASSIGNMENT- (SORTING)

Q1. . Write a program to sort an array in descending order using bubble sort.

Soln.

```java
public class Sorting_Assg1 {

    static void bubbleSort(int[] arr) {
        int n = arr.length;
        for (int i = 0; i < n - 1; i++) {
            for (int j = 0; j < n - i - 1; j++) {
                if (arr[j] < arr[j + 1]) {
                    int temp = arr[j];
                    arr[j] = arr[j + 1];
                    arr[j + 1] = temp;
                }
            }
        }
    }

    public static void main(String[] args) {
        int[] arr = { 3, 5, 1, 6, 0 };
        bubbleSort(arr);
        for (int i : arr) {
            System.out.print(i + " ");
        }
    }
}
```

Q2. WAP to sort an array in descending order using selection sort.

Soln.

```java
public class Sorting_Assg2 {

    static void selectionSort(int[] arr) {
        int n = arr.length;
        for (int i = 0; i < n; i++) {
            int min_idx = i;
            for (int j = i + 1; j < n; j++) {
                if (arr[j] > arr[min_idx]) {
                    min_idx = j;
                }
            }
            int temp = arr[i];
            arr[i] = arr[min_idx];
            arr[min_idx] = temp;
        }
    }

    public static void main(String[] args) {
        int[] arr = { 3, 5, 1, 6, 0 };
        selectionSort(arr);
        for (int i : arr) {
            System.out.print(i + " ");
        }
    }
}
```

Q3. WAP to sort an array in decreasing order using insertion sort.

Soln.

```
public class Sorting_Assg3 {

    static void insertionSort(int[] arr) {
        int n = arr.length;
        for (int i = 1; i < arr.length; i++) {
            int j = i;
            while (j > 0 && arr[j - 1] < arr[j]) {
                int temp = arr[j];
                arr[j] = arr[j - 1];
                arr[j - 1] = temp;
                j--;
            }
        }
    }

    public static void main(String[] args) {
        int[] arr = { 3, 5, 1, 6, 0 };
        insertionSort(arr);
        for (int i : arr) {
            System.out.print(i + " ");
        }
    }
}
```

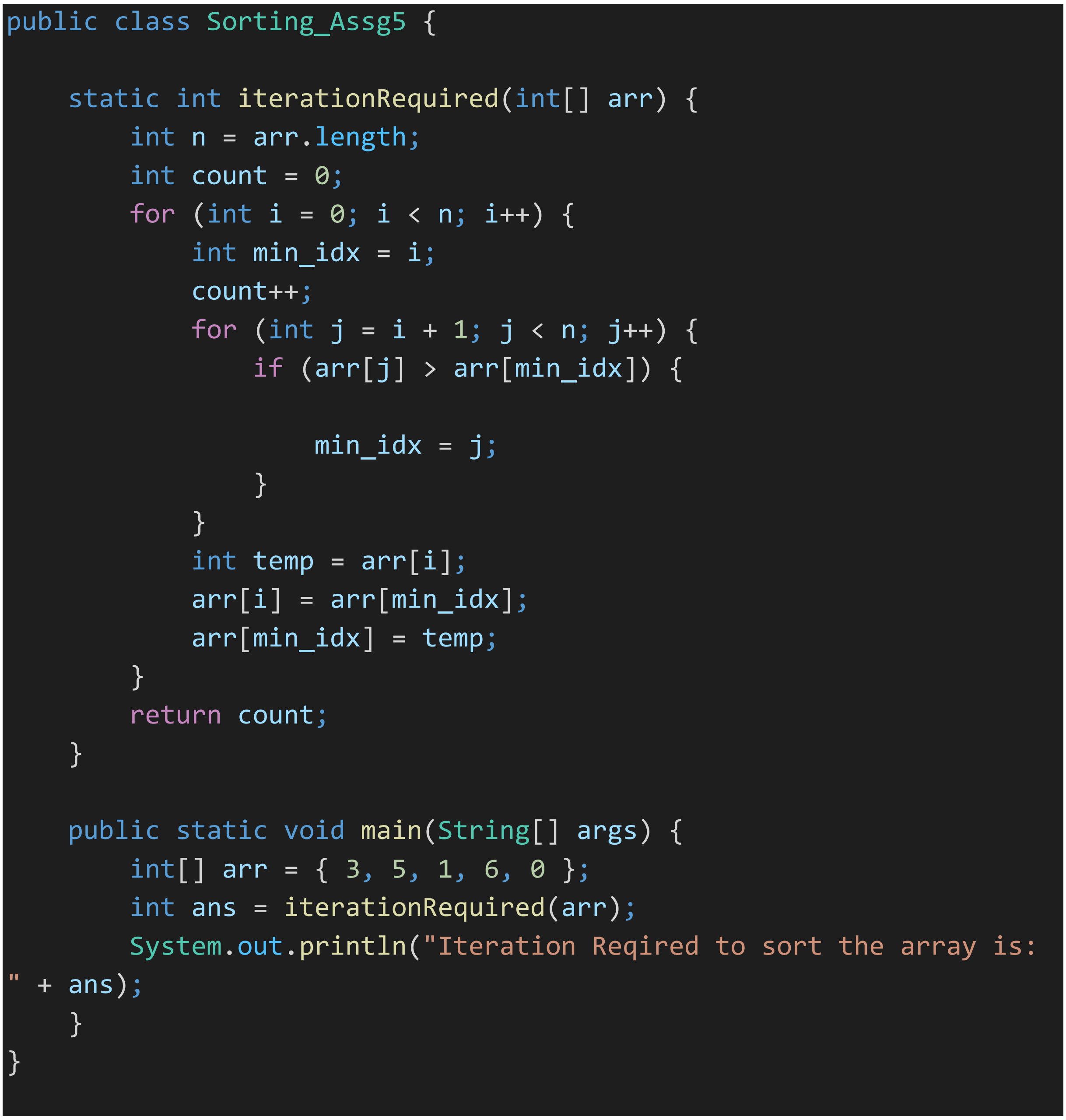
Q4. Find out how many pass would be required to sort the following array in decreasing order using bubble sort.

Soln.

```
public class Sorting_Assg4 {

    static int passRequired(int[] arr) {
        int n = arr.length;
        int count = 0;
        for (int i = 0; i < n - 1; i++) {
            boolean flag = false;
            for (int j = 0; j < n - i - 1; j++) {
                if (arr[j] < arr[j + 1]) {
                    int temp = arr[j];
                    arr[j] = arr[j + 1];
                    arr[j + 1] = temp;
                    flag = true;
                    count++;
                }
            }
            if (!flag) {
                break;
            }
        }
        return count;
    }

    public static void main(String[] args) {
        int[] arr = { 3, 5, 1, 6, 0 };
        int ans = passRequired(arr);

        System.out.println("Passes Required to sort the array are: "
+ ans);
    }
}
```

Q5. Find out the number of iterations to sort the array in descending order using selection sort.

Soln.



```
public class Sorting_Assg5 {

    static int iterationRequired(int[] arr) {
        int n = arr.length;
        int count = 0;
        for (int i = 0; i < n; i++) {
            int min_idx = i;
            count++;
            for (int j = i + 1; j < n; j++) {
                if (arr[j] > arr[min_idx]) {

                    min_idx = j;
                }
            }
            int temp = arr[i];
            arr[i] = arr[min_idx];
            arr[min_idx] = temp;
        }
        return count;
    }

    public static void main(String[] args) {
        int[] arr = { 3, 5, 1, 6, 0 };
        int ans = iterationRequired(arr);
        System.out.println("Iteration Reqired to sort the array is:
" + ans);
    }
}
```