

Assignment Questions

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

Input Array {3,5,1,6,0}

Output Array: {6, 5, 3, 1, 0}

```
import java.io.*;
import java.util.*;
public class Sort {
    // 0 based indexing used
    public static void bubbleSort(int[] a) {
        int n = a.length;

        for (int i = 0; i < n; i++) {
            boolean flag = false;

            for (int j = 0; j < n - i - 1; j++) {
                if (a[j] < a[j + 1]) {
                    flag = true;
                    // swap the values of a[j] and a[j+1]
                    int temp = a[j];
                    a[j] = a[j + 1];
                    a[j + 1] = temp;
                }
            }
            // No Swapping happened, array is sorted
            if (!flag) {
                return;
            }
        }
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the size of array");
        int n = sc.nextInt();
        int[] arr = new int[n];
        System.out.println("Enter the elements of array");
        for (int i = 0; i < n; i++) {
            arr[i] = sc.nextInt();
        }
        bubbleSort(arr);
        for (int i = 0; i < n; i++) {
            System.out.print(arr[i] + " ");
        }
    }
}
```

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

Input Array {3,5,1,6,0}

Output Array: {6, 5, 3, 1, 0}

```
import java.io.*;
import java.util.*;

public class Sort {
    // 0 based indexing used
    public static void selectionSort(int[] a) {
        int n = a.length;

        for (int i = 0; i < n - 1; i++)
            // i represents the current index
            {

                // Find the maximum element in unsorted part of the array
                int max_index = i;
                for (int j = i + 1; j < n; j++) {
                    if (a[j] > a[max_index])
                        max_index = j;
                }

                // Swap the found maximum element with the current element
                if (max_index != i) {
                    int temp = a[max_index];
                    a[max_index] = a[i];
                    a[i] = temp;
                }
            }
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the size of array");
        int n = sc.nextInt();
        int[] arr = new int[n];
        System.out.println("Enter the elements of array");
        for (int i = 0; i < n; i++) {
            arr[i] = sc.nextInt();
        }
        selectionSort(arr);
        for (int i = 0; i < n; i++) {
            System.out.print(arr[i] + " ");
        }
        System.out.print("\n");
    }
}
```

```
}
```

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

Input Array {3,5,1,6,0}

Output Array: {6, 5, 3, 1, 0}

```
// 0-based indexing used here

import java.io.*;
import java.util.*;

public class Sort {
    public static void insertionSort(int[] a) {
        int n = a.length;

        for (int i = 1; i < n; i++) {
            int j = i;

            // Insert a[i] into sorted left part 0..i-1
            while (j > 0 && a[j] > a[j - 1]) {

                // Swap a[j] and a[j-1]
                int temp = a[j];
                a[j] = a[j - 1];
                a[j - 1] = temp;

                // Decrement j by 1
                j--;
            }
        }
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the size of array");
        int n = sc.nextInt();
        int[] arr = new int[n];
        System.out.println("Enter the elements of array");
        for (int i = 0; i < n; i++) {
            arr[i] = sc.nextInt();
        }
        insertionSort(arr);
        for (int i = 0; i < n; i++) {
```

```

        System.out.print(arr[i] + " ");
    }
    System.out.print("\n ");
}
}

```

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

Input Array {3,5,1,6,0}

```

import java.io.*;
import java.util.*;
public class Sort {
    // 0 based indexing used
    public static void bubbleSort(int[] a) {
        int n = a.length;

        for (int i = 0; i < n; i++) {
            boolean flag = false;

            for (int j = 0; j < n - i - 1; j++) {
                if (a[j] > a[j + 1]) {
                    flag = true;
                    // swap the values of a[j] and a[j+1]
                    int temp = a[j];
                    a[j] = a[j + 1];
                    a[j + 1] = temp;
                }
            }
            // No Swapping happened, array is sorted
            if (!flag) {
                return;
            }
        }
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the size of array");
        int n = sc.nextInt();
        int[] arr = new int[n];
        System.out.println("Enter the elements of array");
        for (int i = 0; i < n; i++) {
            arr[i] = sc.nextInt();
        }
        bubbleSort(arr);
        for (int i = 0; i < n; i++) {

```

```

        System.out.print(arr[i] + " ");
    }
}
}

```

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

Input Array {3,5,1,6,0}

```

import java.io.*;
import java.util.*;

public class Sort {
    // 0 based indexing used
    public static void selectionSort(int[] a) {
        int n = a.length;

        for (int i = 0; i < n - 1; i++)
            // i represents the current index
            {

                // Find the minimum element in unsorted part of the array
                int min_index = i;
                for (int j = i + 1; j < n; j++) {
                    if (a[j] < a[min_index])
                        min_index = j;
                }

                // Swap the found minimum element with the current element
                if (min_index != i) {
                    int temp = a[min_index];
                    a[min_index] = a[i];
                    a[i] = temp;
                }
            }
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the size of array");
        int n = sc.nextInt();
        int[] arr = new int[n];
        System.out.println("Enter the elements of array");
        for (int i = 0; i < n; i++) {
            arr[i] = sc.nextInt();
        }
    }
}

```

```
    }  
    selectionSort(arr);  
    for (int i = 0; i < n; i++) {  
        System.out.print(arr[i] + " ");  
    }  
}  
  
}
```