

```

1  /* PRACTICAL-7: Implementation of Sorting techniques.
2  (a) Bubble Sort
3  (b) Selection Sort
4  (c) Merge Sort.
5
6  Name: Angat Shah
7  Enrollment No: 202203103510097
8  Branch: B.Tech Computer Science and Engineering */
9
10 // CODE:
11
12 import java.util.Scanner;
13
14 public class practical7 {
15
16     // I. Bubble Sort
17     public static void bubbleSort(int[] arr) {
18         int n = arr.length;
19         boolean swapped;
20         for (int i = 0; i < n - 1; i++) {
21             swapped = false;
22             for (int j = 0; j < n - i - 1; j++) {
23                 if (arr[j] > arr[j + 1]) {
24                     int temp = arr[j];
25                     arr[j] = arr[j + 1];
26                     arr[j + 1] = temp;
27                     swapped = true;
28                 }
29             }
30             if (!swapped)
31                 break;
32             System.out.println("Step " + (i + 1) + ": " + arrayToString(arr));
33         }
34     }
35
36     // II. Selection Sort
37     public static void selectionSort(int[] arr) {
38         int n = arr.length;
39         for (int i = 0; i < n - 1; i++) {
40             int minIndex = i;
41             for (int j = i + 1; j < n; j++) {
42                 if (arr[j] < arr[minIndex]) {
43                     minIndex = j;
44                 }
45             }
46             int temp = arr[minIndex];
47             arr[minIndex] = arr[i];
48             arr[i] = temp;
49             System.out.println("Step " + (i + 1) + ": " + arrayToString(arr));
50         }
51     }
52
53     // III. Merge Sort
54     public static void mergeSort(int[] arr, int l, int r) {
55         if (l < r) {
56             int mid = (l + r) / 2;
57             mergeSort(arr, l, mid);
58             mergeSort(arr, mid + 1, r);

```

```

59     merge(arr, l, mid, r);
60 }
61 }
62
63 private static void merge(int[] arr, int l, int mid, int r) {
64     int n1 = mid - l + 1;
65     int n2 = r - mid;
66
67     int[] leftArr = new int[n1];
68     int[] rightArr = new int[n2];
69
70     for (int i = 0; i < n1; i++) {
71         leftArr[i] = arr[l + i];
72     }
73     for (int j = 0; j < n2; j++) {
74         rightArr[j] = arr[mid + 1 + j];
75     }
76
77     int i = 0, j = 0, k = l;
78     while (i < n1 && j < n2) {
79         if (leftArr[i] <= rightArr[j]) {
80             arr[k] = leftArr[i];
81             i++;
82         } else {
83             arr[k] = rightArr[j];
84             j++;
85         }
86         k++;
87     }
88
89     while (i < n1) {
90         arr[k] = leftArr[i];
91         i++;
92         k++;
93     }
94
95     while (j < n2) {
96         arr[k] = rightArr[j];
97         j++;
98         k++;
99     }
100 }
101
102 private static String arrayToString(int[] arr) {
103     StringBuilder sb = new StringBuilder();
104     for (int num : arr) {
105         sb.append(num).append(" ");
106     }
107     return sb.toString();
108 }
109
110 public static void main(String[] args) {
111     Scanner scanner = new Scanner(System.in);
112
113     System.out.print("Enter the size of the array: ");
114     int size = scanner.nextInt();
115     int[] arr = new int[size];
116

```

```
117 System.out.println("Enter the elements of the array:");
118 for (int i = 0; i < size; i++) {
119     arr[i] = scanner.nextInt();
120 }
121
122 System.out.println("Choose sorting algorithm:");
123 System.out.println("1. Bubble Sort");
124 System.out.println("2. Selection Sort");
125 System.out.println("3. Merge Sort");
126 System.out.print("Enter your choice: ");
127 int choice = scanner.nextInt();
128
129 switch (choice) {
130     case 1:
131         System.out.println("\nApplying Bubble Sort:");
132         bubbleSort(arr);
133         break;
134     case 2:
135         System.out.println("\nApplying Selection Sort:");
136         selectionSort(arr);
137         break;
138     case 3:
139         System.out.println("\nApplying Merge Sort:");
140         mergeSort(arr, 0, arr.length - 1);
141         break;
142     default:
143         System.out.println("Invalid choice!");
144 }
145 System.out.println("\nSorted Array:");
146 System.out.println(arrayToString(arr));
147
148 scanner.close();
149 }
150 }
```