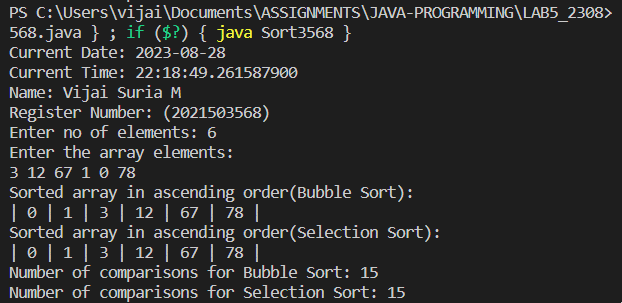
|  |
| --- |
| Exp: 05 Date: 28/08/2023  JAVA PROGRAMMING LAB 5 |

Name: VIJAI SURIA M

Reg No.: 2021503568

1. **Exercise 1 - Sort Write a program to read n integer in a 1D array and print the sorted array in the following format. Use static methods and find the number of comparisons for the sorting algorithm whose worst-case complexity is O(n^2) and O(n)**

|  |
| --- |
| import java.time.LocalDate;  import java.time.LocalTime;  import java.util.Scanner;  public class Sort3568 {      public static void main(String[] args) {          System.out.println("Current Date: " + LocalDate.now());          System.out.println("Current Time: " + LocalTime.now());          System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");          Scanner scanner = new Scanner(System.in);          System.out.print("Enter no of elements: ");          int n = scanner.nextInt();          int[] arr = new int[n];          System.out.println("Enter the array elements: ");          for (int i = 0; i < n; i++) {              arr[i] = scanner.nextInt();          }          int comparisonsBubbleSort = bubbleSort(arr.clone());          int comparisonsSelectionSort = selectionSort(arr.clone());          System.out.println("Number of comparisons for Bubble Sort: " + comparisonsBubbleSort);          System.out.println("Number of comparisons for Selection Sort: " + comparisonsSelectionSort);      }      public static int bubbleSort(int[] arr) {          int n = arr.length;          int comparisons = 0;          for (int i = 0; i < n - 1; i++) {              for (int j = 0; j < n - i - 1; j++) {                  comparisons++;                  if (arr[j] > arr[j + 1]) {                      int temp = arr[j];                      arr[j] = arr[j + 1];                      arr[j + 1] = temp;                  }              }          }          System.out.println("Sorted array in ascending order(Bubble Sort):");          printArray(arr);          return comparisons;      }      public static int selectionSort(int[] arr) {          int n = arr.length;          int comparisons = 0;          for (int i = 0; i < n - 1; i++) {              int minIndex = i;              for (int j = i + 1; j < n; j++) {                  comparisons++;                  if (arr[j] < arr[minIndex]) {                      minIndex = j;                  }              }              int temp = arr[minIndex];              arr[minIndex] = arr[i];              arr[i] = temp;          }          System.out.println("Sorted array in ascending order(Selection Sort):");          printArray(arr);          return comparisons;      }      public static void printArray(int[] arr) {          System.out.print("|");          for (int i = 0; i < arr.length; i++) {              System.out.printf(" %d |", arr[i]);          }          System.out.println();      }  } |

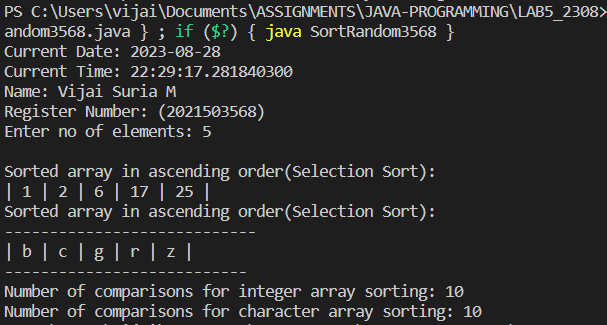


1. **Exercise 2: Sort random integer/character Write a program to read n random integer in a 1D array.**

**a) Apply method to sort the generated array content and return the number of comparisons done.**

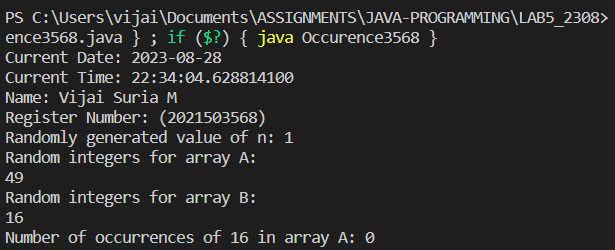
**b) Apply another method to generate character array using the random integer and sort the array.**

|  |
| --- |
| import java.time.LocalDate;  import java.time.LocalTime;  import java.util.Random;  import java.util.Scanner;  public class SortRandom3568 {      public static void main(String[] args) {          System.out.println("Current Date: " + LocalDate.now());          System.out.println("Current Time: " + LocalTime.now());          System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");          Scanner scanner = new Scanner(System.in);          System.out.print("Enter no of elements: ");          int n = scanner.nextInt();          int[] arr = new int[n];            arr = RandomArray(arr);          int comparisonsInt = OrderNSort(arr.clone());          char[] charArray = CharArray(arr);          int comparisonsChar = OrderNSort(charArray.clone());          System.out.println("Number of comparisons for integer array sorting: " + comparisonsInt);          System.out.println("Number of comparisons for character array sorting: " + comparisonsChar);      }      public static int[] RandomArray(int[] arr) {          Random random = new Random();          for (int i = 0; i < arr.length; i++) {              arr[i] = random.nextInt(26);          }          return arr;      }      public static char[] CharArray(int[] intArray) {          char[] charArray = new char[intArray.length];          for (int i = 0; i < intArray.length; i++) {              charArray[i] = (char) (intArray[i] + 'a');          }          return charArray;      }      public static int OrderNSort(int[] arr) {          int comparisons = 0;          for (int i = 0; i < arr.length - 1; i++) {              for (int j = 0; j < arr.length - i - 1; j++) {                  comparisons++;                  if (arr[j] > arr[j + 1]) {                      int temp = arr[j];                      arr[j] = arr[j + 1];                      arr[j + 1] = temp;                  }              }          }          System.out.println("\nSorted array in ascending order(Selection Sort):");          printArray(arr);          return comparisons;      }      public static int OrderNSort(char[] arr) {          int comparisons = 0;          for (int i = 0; i < arr.length - 1; i++) {              for (int j = 0; j < arr.length - i - 1; j++) {                  comparisons++;                  if (arr[j] > arr[j + 1]) {                      char temp = arr[j];                      arr[j] = arr[j + 1];                      arr[j + 1] = temp;                  }              }          }          System.out.println("\nSorted array in ascending order(Selection Sort):");          printArray(arr);          return comparisons;      }      public static void printArray(int[] arr) {          System.out.print("|");          for (int i = 0; i < arr.length; i++) {              System.out.printf(" %d |", arr[i]);          }      }      public static void printArray(char[] arr) {          System.out.print("----------------------------\n");          System.out.print("|");          for (int i = 0; i < arr.length; i++) {              System.out.printf(" %c |", arr[i]);          }          System.out.print("\n---------------------------\n");      }  } |



1. **Exercise 3: Search element Occurrence Write a program to read n random integer in a 1D array of A and B of size n. Apply method to search the occurrence of element in B and print the number of B element occurrence in A.**

|  |
| --- |
| import java.time.LocalDate;  import java.time.LocalTime;  import java.util.Random;  import java.util.Scanner;  public class SortRandom3568 {      public static void main(String[] args) {          System.out.println("Current Date: " + LocalDate.now());          System.out.println("Current Time: " + LocalTime.now());          System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");          Scanner scanner = new Scanner(System.in);          System.out.print("Enter no of elements: ");          int n = scanner.nextInt();          int[] arr = new int[n];            arr = RandomArray(arr);          int comparisonsInt = OrderNSort(arr.clone());          char[] charArray = CharArray(arr);          int comparisonsChar = OrderNSort(charArray.clone());          System.out.println("Number of comparisons for integer array sorting: " + comparisonsInt);          System.out.println("Number of comparisons for character array sorting: " + comparisonsChar);      }      public static int[] RandomArray(int[] arr) {          Random random = new Random();          for (int i = 0; i < arr.length; i++) {              arr[i] = random.nextInt(26);          }          return arr;      }      public static char[] CharArray(int[] intArray) {          char[] charArray = new char[intArray.length];          for (int i = 0; i < intArray.length; i++) {              charArray[i] = (char) (intArray[i] + 'a');          }          return charArray;      }      public static int OrderNSort(int[] arr) {          int comparisons = 0;          for (int i = 0; i < arr.length - 1; i++) {              for (int j = 0; j < arr.length - i - 1; j++) {                  comparisons++;                  if (arr[j] > arr[j + 1]) {                      int temp = arr[j];                      arr[j] = arr[j + 1];                      arr[j + 1] = temp;                  }              }          }          System.out.println("\nSorted array in ascending order(Selection Sort):");          printArray(arr);          return comparisons;      }      public static int OrderNSort(char[] arr) {          int comparisons = 0;          for (int i = 0; i < arr.length - 1; i++) {              for (int j = 0; j < arr.length - i - 1; j++) {                  comparisons++;                  if (arr[j] > arr[j + 1]) {                      char temp = arr[j];                      arr[j] = arr[j + 1];                      arr[j + 1] = temp;                  }              }          }          System.out.println("\nSorted array in ascending order(Selection Sort):");          printArray(arr);          return comparisons;      }      public static void printArray(int[] arr) {          System.out.print("|");          for (int i = 0; i < arr.length; i++) {              System.out.printf(" %d |", arr[i]);          }      }      public static void printArray(char[] arr) {          System.out.print("----------------------------\n");          System.out.print("|");          for (int i = 0; i < arr.length; i++) {              System.out.printf(" %c |", arr[i]);          }          System.out.print("\n---------------------------\n");      }  } |



1. **Exercise 4: Sum of arrays Write a program to read two 2D array. Apply method to perform column major sum and sort the array based on the sum of columns.**