Code for Quick Sort:

import java.util.\*;

public class EXPT\_03\_QuickSort {

    int sub\_problem\_processing(int arr[], int start, int end) {

        int pivot = arr[end];

        int i = (start - 1);

        for (int j = start; j <= end - 1; j++) {

            // if current element is smaller than pivot

            if (arr[j] < pivot) {

                i++; // increment index of smaller element

                int temp = arr[i];

                arr[i] = arr[j];

                arr[j] = temp;

            }

        }

        int temp1 = arr[i + 1];

        arr[i + 1] = arr[end];

        arr[end] = temp1;

        return (i + 1);

    }

    void sub\_problem\_definition(int arr[], int start, int end) {

        if (start < end) {

            int p = sub\_problem\_processing(arr, start, end); // p is partitioning index

            sub\_problem\_definition(arr, start, p - 1);

            sub\_problem\_definition(arr, p + 1, end); // divide and conquer approach

        }

    }

    public static void main(String args[]) {

        Scanner in = new Scanner(System.in);

        System.out.print("Enter the number of elements for array: ");

        int n = in.nextInt();

        int arr[] = new int[n];

        Random randomnumber = new Random();

        System.out.print("\nThe randomly-generated array is ");

        for (int i = 0; i < n; i++) {

            arr[i] = randomnumber.nextInt(1000);

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

        }

        EXPT\_03\_QuickSort obj = new EXPT\_03\_QuickSort();

        long start = System.currentTimeMillis();

        obj.sub\_problem\_definition(arr, 0, n - 1);

        long end = System.currentTimeMillis();

        long time = end - start;

        System.out.print("\nAfter sorting by Quick Sort, the array elements are ");

        for (int i = 0; i < n; i++) {

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

        }

        System.out.print("\nThe number of elements in array are " + n + ".");

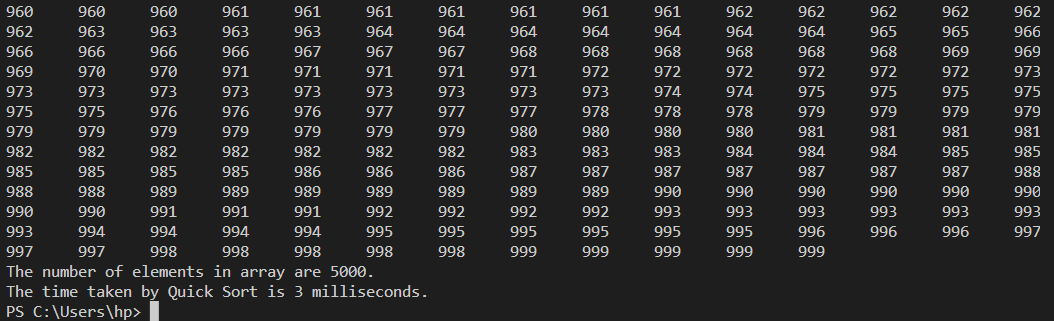
        System.out.print("\nThe time taken by Quick Sort is " + time + " milliseconds.");

    }

}

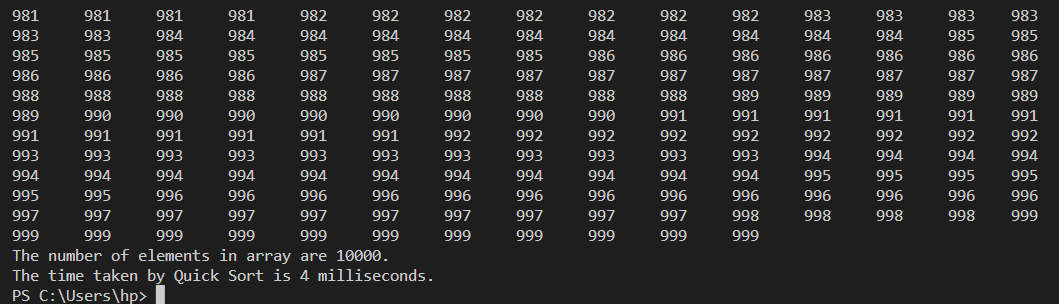
Case 1: When the number of elements is 5,000

3 milliseconds



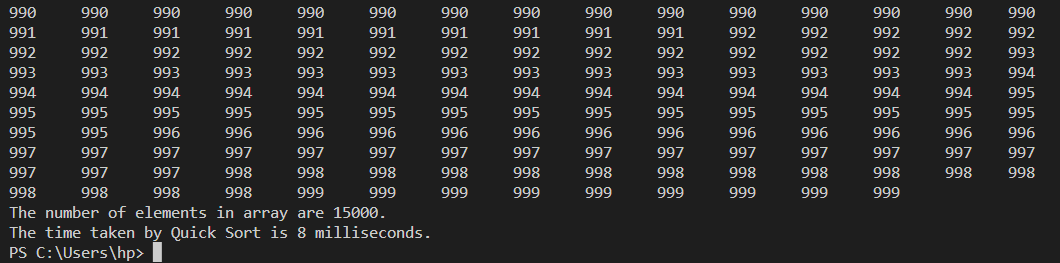
Case 2: When the number of elements is 10,000

4 milliseconds



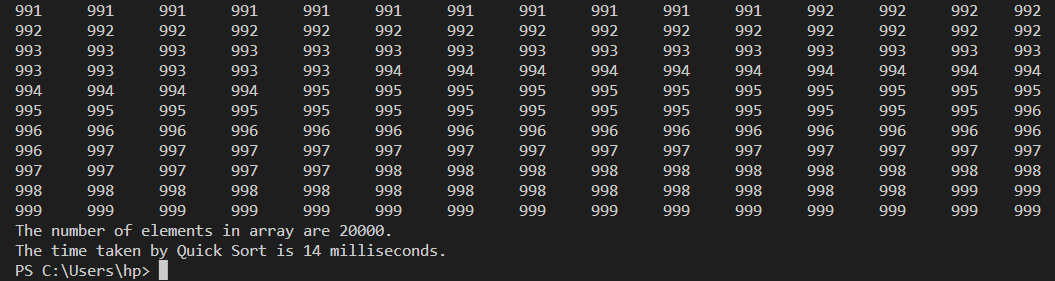
Case 3: When the number of elements is 15,000

8 milliseconds



Case 4: When the number of elements is 20,000

14 milliseconds



Case 5: When the number of elements is 25,000

20 milliseconds

