import java.util.Arrays; // Importa la clase Arrays para manipular y mostrar el contenido de arreglos

import java.util.Random; // Importa la clase Random para generar números aleatorios

public class App {

    public static void main(String[] args) throws Exception {

        // Declara un arreglo de enteros de tamaño 10

        int[] array = new int[10];

        // Llena el arreglo con números aleatorios entre 1 y 10

        for (int i = 0; i < array.length; i++) {

            Random random = new Random(); // Crea una nueva instancia de Random en cada iteración

            int x = random.nextInt(10) + 1; // Genera un número aleatorio entre 1 y 10 (inclusive)

            array[i] = x; // Asigna el número aleatorio generado al índice i del arreglo

        }

        // Imprime el arreglo original

        System.out.println("Original");

        System.out.println(Arrays.toString(array)); // Convierte el arreglo a una cadena y lo imprime

        // Inicia las variables para el proceso de reorganización

        int j = 0; // Variable para rastrear el índice de intercambio

        int termina = array.length - 1; // Variable que guarda el último índice del arreglo

        // Bucle principal para recorrer el arreglo y hacer intercambios

        for (int i = 0; i <= termina; i++) {

            // Si el valor en la posición j es menor o igual al valor en la posición i

            if (array[j] <= array[i]) {

                // Intercambia los valores en las posiciones i y j

                int temporal = array[i]; // Guarda el valor en array[i] temporalmente

                array[i] = array[j]; // Asigna el valor en array[j] a array[i]

                array[j] = temporal; // Asigna el valor temporal a array[j]

                j++; // Incrementa el índice j

                // Bucle para buscar y ordenar los elementos según una condición interna

                for (int x = 0; x <= termina; x++) {

                    if (array[i] <= array[x]) { // Compara el valor actual con el resto de los elementos

                        // Intercambia los valores si se cumple la condición

                        temporal = array[x]; // Guarda temporalmente el valor de array[x]

                        array[x] = array[i]; // Asigna el valor de array[i] a array[x]

                        array[i] = temporal; // Asigna el valor temporal a array[i]

                    }

                }

            } else {

                j++; // Si la condición inicial no se cumple, incrementa j para pasar al siguiente valor

            }

            // Imprime el estado actual del arreglo después de cada iteración

            System.out.println(Arrays.toString(array));

        }

    }

}