7) Hacer un programa que dado un arreglo ordenado creciente de enteros

de tamaño 10 que se encuentra precargado, solicite al usuario un numero

entero y lo inserte en el arreglo manteniendo su orden. Para ello tendrá

que realizar un corrimiento a derecha (se pierde el último valor del arreglo)

y colocar el numero en el arreglo en la posición indicada.

import java.util.Random;

import java.io.BufferedReader;

import java.io.InputStreamReader;

public class carole7 {

    public static int MAX = 10;

    public static final int MAXVALOR = 9;

    public static final int MINVALOR = 1;

    public static void main(String[] args) {

        int[] arrenteros = new int[MAX];

        int pos, num;

        BufferedReader entrada = new BufferedReader(new InputStreamReader(System.in));

        cargar\_arreglo\_aleatorio\_int(arrenteros);

        ordenar\_arreglo\_burbujeo(arrenteros);

        imprimir\_arreglo\_int(arrenteros);

        try {

            System.out.println("ingrese un numero entero");

            num = Integer.valueOf(entrada.readLine());

            pos = obtener\_pos\_arreglo\_creciente(arrenteros, num);

            insertar\_num(arrenteros, pos, num);

            imprimir\_arreglo\_int(arrenteros);

        } catch (Exception e) {

            // TODO: handle exception

        }

    }

    public static void insertar\_num(int[] arr, int p, int numero) {

        corrimiento\_der(arr, p);

        arr[p] = numero;

    }

    public static void corrimiento\_der(int[] arr, int p) {

        int i = MAX - 1;

        while (i > p) {

            arr[i] = arr[i - 1];

            p--;

        }

    }

    public static int obtener\_pos\_arreglo\_creciente(int[] arr, int numero) {

        int posicion = 0;

        while ((posicion < MAX) && (arr[posicion] < numero)) {

            posicion++;

        }

        return posicion;

    }

    public static void ordenar\_arreglo\_burbujeo(int[] arr) {

        int temp;

        for (int i = 1; i < MAX; i++) {

            for (int j = 0; j < MAX - 1; j++) {

                if (arr[j] > arr[j + 1]) {

                    temp = arr[j];

                    arr[j] = arr[j + 1];

                    arr[j + 1] = temp;

                }

            }

        }

    }

8) Hacer un programa que dado un arreglo ordenado creciente de enteros

de tamaño 10 que se encuentra precargado, solicite al usuario un numero

entero y elimine la primer ocurrencia de numero (un número igual) en el

arreglo (si existe).

import java.util.Random;

import java.io.BufferedReader;

import java.io.InputStreamReader;

public class carole8 {

    public static int MAX = 10;

    public static final int MAXVALOR = 9;

    public static final int MINVALOR = 1;

    public static void main(String[] args) {

        int[] arrenteros = new int[MAX];

        int pos, num;

        BufferedReader entrada = new BufferedReader(new InputStreamReader(System.in));

        cargar\_arreglo\_aleatorio\_int(arrenteros);

        ordenar\_arreglo\_burbujeo(arrenteros);

        imprimir\_arreglo\_int(arrenteros);

        try {

            System.out.println("ingrese un numero entero");

            num = Integer.valueOf(entrada.readLine());

            pos = obtener\_pos\_arreglo\_creciente(arrenteros, num);

            corrimeinto\_izq(arrenteros, pos);

            imprimir\_arreglo\_int(arrenteros);

        } catch (Exception e) {

            // TODO: handle exception

        }

    }

    public static void corrimeinto\_izq(int[] arr, int p) {

        while (p < MAX - 1) {

            arr[p] = arr[p + 1];

            p++;

        }

    }

    public static int obtener\_pos\_arreglo\_creciente(int[] arr, int n) {

        int posicion = 0;

        while ((posicion < MAX) && (arr[posicion] < n)) {

            posicion++;

        }

        return posicion;

    }

    public static void ordenar\_arreglo\_burbujeo(int[] arr) {

        int temp;

        for (int i = 1; i < MAX; i++) {

            for (int j = 0; j < MAX - 1; j++) {

                if (arr[j] > arr[j + 1]) {

                    temp = arr[j];

                    arr[j] = arr[j + 1];

                    arr[j + 1] = temp;

                }

            }

        }

    }

9)discutir con carola.

10) Se tiene un arreglo de enteros de tamaño 20 de secuencias de números

entre 1 y 9, separadas por 0. El arreglo esta precargado, y además

empieza y termina con uno o más separadores 0. Hacer un programa que

permita obtener a través de métodos la posición de inicio y la posición de

fin de la secuencia ubicada a partir de una posición entera ingresada por

el usuario. Finalmente, si existen imprima por pantalla ambas posiciones

obtenidas.

import java.util.Random;

import java.io.BufferedReader;

import java.io.InputStreamReader;

public class carole10 {

    public static final int MAX = 20;

    public static final int MAXVALOR = 9;

    public static final int MINVALOR = 1;

    public static final double probabilidad\_numero = 0.4;

    public static void main(String[] args) {

        int[] arrenteros = new int[MAX];

        int pos, inicio, fin;

        BufferedReader entrada = new BufferedReader(new InputStreamReader(System.in));

        cargar\_arreglo\_aleatorio\_secuencias\_int(arrenteros);

        imprimir\_arreglo\_secuencias\_int(arrenteros);

        try {

            System.out.println("ingrese una posicion");

            pos = Integer.valueOf(entrada.readLine());

            inicio = buscar\_ini\_sec(arrenteros, pos);

            fin = buscar\_fin\_sec(arrenteros, inicio);

            System.out.println("el ini esta en " + inicio + "el fin en " + fin);

        } catch (Exception e) {

            // TODO: handle exception

        }

    }

    public static int buscar\_ini\_sec(int[] arr, int p) {

        while ((p < MAX) && (arr[p] == 0)) {

            p++;

        }

        return p;

    }

    public static int buscar\_fin\_sec(int[] arr, int p) {

        while ((p < MAX) && (arr[p] != 0)) {

            p++;

        }

        return p - 1;

    }