1- Realiza la función: int[] buscarTodos(int t[], int clave), que crea y devuelve una tabla con todos los índices de los elementos donde se encuentra la clave de búsqueda. En el caso de que clave no se encuentre en la tabla t, la función devolverá una tabla vacía.

package main;

import java.util.Arrays;

import java.util.Random;

import java.util.Scanner;

public class Main {

static Random rand = new Random();

Scanner sc = new Scanner(System.in);

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Escribe la longitud del array y la clave");

int lon = sc.nextInt();

int clave = sc.nextInt();

int[] t = new int[lon];

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

t[i] = rand.nextInt(1, 5); // rellena el array con números random

}

int[] busClave = buscar(t, clave);

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

}

public static int[] buscar(int[] t, int clave) {

boolean noExiste = true;

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

if (t[i] == clave) {

System.out.println("La clave se encuentra en " + i);

noExiste = false;

}

}

if (noExiste == true) {

Arrays.fill(t, 0);

}

return t;

}

}

2- Escribe la función void desordenar(int t[]), que cambia de forma aleatoria los elementos contenidos en la tabla t. Si la tabla estuviera ordenada, dejaría de estarlo

import java.util.Random;

public class Desordenar {

public static void desordenar(int[] t) {

Random random = new Random();

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

int j = random.nextInt(t.length); // Elige un índice aleatorio

// Intercambia los elementos

int temp = t[i];

t[i] = t[j];

t[j] = temp;

}

}

}

3- Modifica la actividad anterior para que la función no modifique la tabla que se pasa como parámetro y, en su lugar, cree y devuelva una copia de la tabla donde se han desordenado los valores de los elementos.

import java.util.Random;

public class Desordenar {

public static int[] desordenar(int[] t) {

Random random = new Random();

int[] copia = new int[t.length]; // Crear un nuevo array

System.arraycopy(t, 0, copia, 0, t.length); // Copiar el contenido de t a copia

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

int j = random.nextInt(copia.length); // Elige un índice aleatorio

// Intercambia los elementos

int temp = copia[i];

copia[i] = copia[j];

copia[j] = temp;

}

return copia;

}

}

4- El ayuntamiento de tu localidad te ha encargado una aplicación que ayude a realizar encuestas estadísticas para conocer el nivel adquisitivo de los habitantes del municipio. Para ello, tendrás que preguntar el sueldo a cada persona encuestada. A priori, no conoces el número de encuestados. Para finalizar la entrada de datos, introduce un sueldo con valor -1. Una vez terminada la entrada de datos, muestra la siguiente información: ✓ Todos los sueldos introducidos ordenados de forma decreciente. ✓ El sueldo máximo y mínimo. ✓ La media de los sueldos.

import java.util.Arrays;

import java.util.Scanner;

public class Encuesta {

public static void main(String[] args) {

int[] sueldos = new int[10]; // Suponemos que hay un máximo de 10 sueldos

int contador = 0;

Scanner scanner = new Scanner(System.in);

while (true) {

System.out.print("Introduce el sueldo (o -1 para terminar): ");

int sueldo = scanner.nextInt();

if (sueldo == -1) {

break;

}

// Si el arreglo está lleno, duplicamos el tamaño

if (contador == sueldos.length) {

sueldos = Arrays.copyOf(sueldos, sueldos.length \* 2);

}

sueldos[contador] = sueldo;

contador++;

}

// Ordenar sueldos de forma decreciente

Arrays.sort(sueldos, 0, contador);

for (int i = 0; i < contador / 2; i++) {

int temp = sueldos[i];

sueldos[i] = sueldos[contador - 1 - i];

sueldos[contador - 1 - i] = temp;

}

// Mostrar los sueldos ordenados

System.out.println("Sueldos ordenados de forma decreciente: ");

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

System.out.print(sueldos[i] + " ");

}

System.out.println();

// Mostrar sueldo máximo y mínimo

int maxSueldo = sueldos[0];

int minSueldo = sueldos[contador - 1];

System.out.println("Sueldo máximo: " + maxSueldo);

System.out.println("Sueldo mínimo: " + minSueldo);

// Calcular la media

double suma = 0;

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

suma += sueldos[i];

}

double media = suma / contador;

System.out.println("La media de los sueldos es: " + media);

}

}

5- La fusión de dos tablas ordenadas consiste en copiar todos sus elementos (de ambas tablas) en una tercera que deberá seguir ordenada. Podemos realizar una fusión «ineficiente» copiando los elementos de ambas tablas (sin tener en cuenta el orden) en la tabla final y ordenar esta.

import java.util.Arrays;

public class Fusion {

public static int[] fusion(int[] t1, int[] t2) {

// Crear un array para la fusión

int[] fusionada = new int[t1.length + t2.length];

// Copiar los elementos de t1 y t2

System.arraycopy(t1, 0, fusionada, 0, t1.length);

System.arraycopy(t2, 0, fusionada, t1.length, t2.length);

// Ordenar el array fusionado

Arrays.sort(fusionada);

return fusionada;

}

}

6- Implementa la función: int[] suma(int t[], int numElementos), que crea y devuelve una tabla con las sumas de los numElementos elementos consecutivos de t. Veamos un ejemplo, sea t = [10, 1, 5, 8, 9, 2]. Si los elementos de t se agrupan de 3 en 3, se harán las sumas: 10 + 1 + 5. Igual a 16. 1 + 5 + 8. Igual a 14. 5 + 8 + 9. Igual a 22. 8 + 9 + 2. Igual a 19. Por lo tanto, la función devolverá una tabla con los resultados: [16, 14, 22, 19].

public class Suma {

public static int[] suma(int[] t, int numElementos) {

// Calcular el número de sumas posibles

int numSumas = t.length - numElementos + 1;

int[] resultado = new int[numSumas];

// Calcular las sumas de los numElementos consecutivos

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

int suma = 0;

// Sumar los elementos consecutivos

for (int j = 0; j < numElementos; j++) {

suma += t[i + j];

}

resultado[i] = suma;

}

return resultado;

}

public static void main(String[] args) {

// Ejemplo de uso

int[] t = {10, 1, 5, 8, 9, 2};

int numElementos = 3;

int[] resultado = suma(t, numElementos);

// Imprimir el resultado

for (int suma : resultado) {

System.out.print(suma + " "); // Imprime: 16 14 22 19

}

}

}