1 - Introduce por teclado un número n; a continuación, solicita al usuario que teclee n números. Realiza la media de los números positivos, la media de los negativos y cuenta el número de ceros introducidos.

**package** main;

**import** java.util.Arrays;

**import** java.util.Scanner;

**public** **class** Main {

**public** **static** **void** main(String[] args) {

Scanner sc = **new** Scanner(System.***in***);

**int** n = sc.nextInt();

**int** numPositivos = 0;

**int** numNegativos = 0;

**int** sumaPositivos = 0;

**int** sumaNegativos = 0;

**int** media = 0;

**int** numCeros = 0;

System.***out***.println("Teclea " + n + " números");

**for** (**int** i = 0; i < n; i++) {

**int** numero = sc.nextInt();

**if** (numero < 0) {

numNegativos++;

sumaNegativos += numero;

media = sumaNegativos / numNegativos;

System.***out***.println("La media de positivos es " + media);

} **else** **if** (numero > 0){

numPositivos++;

sumaPositivos += numero;

media = sumaPositivos / numPositivos;

System.***out***.println("La media de positivos es " + media);

} **else** **if** (numero == 0) {

numCeros++;

System.***out***.println("El número de ceros es " + numCeros);

}

}

}

}

2- . Diseñar un programa que solicite al usuario que introduzca por teclado 5 números decimales. A continuación, mostrar los números en el mismo orden que se han introducido. (Usar un array)

**package** main;

**import** java.util.Arrays;

**import** java.util.Scanner;

**public** **class** Main {

**public** **static** **void** main(String[] args) {

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Teclee 5 decimales");

**float**[] numeros = **new** **float** [5];

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

numeros[i] = sc.nextFloat();

}

System.***out***.println(Arrays.*toString*(numeros));

}}

3- Escribir una aplicación que solicite al usuario cuántos números desea introducir. A continuación, introducir por teclado esa cantidad de números enteros, y por último, mostrar en el orden inverso al introducido.

**package** main;

**import** java.util.Arrays;

**import** java.util.Scanner;

**public** **class** Main {

**public** **static** **void** main(String[] args) {

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("¿Cuántos números quieres?");

**int** num = sc.nextInt();

**int**[] numeros = **new** **int** [num];

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

numeros[i] = sc.nextInt();

}

**for** (**int** j = numeros.length - 1; j > 0; j--) { // recorrer desde la última posición hacia delante para invertir números

System.***out***.println("Invirtiendo orden... " + (numeros[j]));

}

}

}

4- Solicitar al usuario la cantidad de números enteros que va a introducir, después el usuario introducirá la cantidad de números indicada y se mostrará por pantalla el número más grande y el más pequeño.

**package** main;

**import** java.util.Arrays;

**import** java.util.Scanner;

**public** **class** Main {

**public** **static** **void** main(String[] args) {

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("¿Cuántos números quieres?");

**int** num = sc.nextInt();

**int**[] numeros = **new** **int** [num];

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

numeros[i] = sc.nextInt();

}

Arrays.*sort*(numeros);

**int** numMayor = numeros[numeros.length - 1];

**int** numMenor = numeros[0];

System.***out***.println("El número mayor es " + numMayor + " y el menor " + numMenor);

}

}

5- Escribir la función public static int[] rellenaPares(int longitud, int fin), que crea y devuelve una tabla ordenada de la longitud especificada, que se encuentra rellena con números pares aleatorios comprendidos en el rango desde 2 hasta fin (inclusive).

**package** main;

**import** java.util.Arrays;

**import** java.util.Random;

**import** java.util.Scanner;

**public** **class** Main {

**public** **static** **void** main(String[] args) {

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Introduce la longitud y el valor superior del rango");

**int** longitud = sc.nextInt();

**int** fin = sc.nextInt();

**int**[] resultado = *rellenaPares*(longitud, fin);

System.***out***.println(Arrays.*toString*(resultado));

}

**public** **static** **int**[] rellenaPares(**int** longitud, **int** fin) {

Random rand = **new** Random();

**int**[] arr = **new** **int** [longitud];

**for** (**int** i = 0; i < longitud; i++) {

**int** n = rand.nextInt(2, fin);

arr[i] = n;

**while** (n % 2 != 0) {

n = rand.nextInt(2, fin);

}

}

Arrays.*sort*(arr);

**return** arr;

}

}

6- Escribe la función: public static int buscar(int t[], int clave), que busca de forma secuencial en la tabla t el valor clave. En caso de encontrarlo, devuelve en qué posición lo encuentra; y en caso contrario, devolverá -1.

**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(busClave);

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

}

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

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

**if** (t[i] == clave) {

**return** i; // Devuelve la posición en la que se encuentra la clave

}

}

**return** -1; // Si no se encuentra la clave, devuelve -1

}

}

7- Implementar la función: public static int[] sinRepetidos(int t[]), que construye y devuelve una tabla de la longitud apropiada, con los elementos de t, donde se han eliminado los datos repetidos.

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");

int lon = sc.nextInt();

int[] t = new int[lon];

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

t[i] = rand.nextInt(1, 4); // rellena el array con números random y ordenamos

}

Arrays.sort(t);

int tAlmacen [] = new int [lon];

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

System.out.println(Arrays.toString(sinRepetidos(t))); // llamamos al array original y a la función

}

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

int [] tAlmacen = new int [0]; // creamos el array donde se guardan los no repes

for (int valor: t) { // recorremos el array original y guardamos cada valor en una variable

if (Arrays.binarySearch(tAlmacen, valor) < 0) { // buscamos binariamente

tAlmacen = Arrays.copyOf(tAlmacen, tAlmacen.length + 1); // como estamos en un bucle, en cada vuelta longitud +1

tAlmacen[tAlmacen.length - 1] = valor; // guardamos en la última posición el nuevo valor

}

}

return tAlmacen;

}

}

8- Leer y almacenar n números enteros en una tabla, a partir de la que se construirán otras dos tablas con los elementos con valores pares e impares de la primera, respectivamente. Las tablas pares e impares deben mostrarse ordenadas.

package main;

import java.util.Arrays;

import java.util.Random;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Escribe la longitud de la tabla");

int lon = sc.nextInt();

int[] t = new int[lon];

int[] tPares = new int[0];

int[] tImpares = new int[0];

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

System.out.println("Escribe números");

t[i] = sc.nextInt();

}

paresImpares(t);

}

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

Scanner sc = new Scanner(System.in);

int[] tPares = new int[0];

int[] tImpares = new int[0];

for (int valor : t) {

if (valor % 2 == 0) {

tPares = Arrays.copyOf(tPares, tPares.length + 1);

tPares[tPares.length - 1] = valor;

} else {

tImpares = Arrays.copyOf(tImpares, tImpares.length + 1);

tImpares[tImpares.length - 1] = valor;

}

}

Arrays.sort(tImpares);

Arrays.sort(tPares);

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

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

return t;

}

}

9- Escribir la función: public static int[] eliminarMayores(int t[], int valor) que crea y devuelve una copia de la tabla t donde se han eliminado todos los elementos que son mayores que valor.

package main;

import java.io.InputStream;

import java.util.Arrays;

import java.util.Random;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Random rand = new Random();

Scanner sc = new Scanner(System.in);

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

int lon = sc.nextInt();

int [] t = new int [lon];

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

t[i] = rand.nextInt(1, 11);

}

Arrays.sort(t);

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

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

}

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

Scanner sc = new Scanner(System.in);

System.out.println("Escribe el valor a buscar");

int valor = sc.nextInt();

int pos = Arrays.binarySearch(t, valor);

if (pos >= 0) { // buscamos si el valor está en el array

System.out.println(pos);

} else {

System.out.println("No se ha encontrado nada");

}

int [] copia = Arrays.copyOfRange(t, 0, pos + 1); // creamos una tabla desde el inicio hasta el valor, de forma que los mayores quedan eliminados

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

return copia;

}

}

10- Desarrollar el juego “la cámara secreta”, que consiste en abrir una cámara mediante su combinación secreta, que está formado por una combinación de dígitos del uno al cinco. El jugador especificará cuál es la longitud de la combinación; a mayor longitud, mayor será la dificultad del juego. La aplicación genera, de forma aleatoria, una combinación secreta que el usuario tendrá que acertar. En cada intento se muestra como pista, para cada dígito de la combinación introducido por el jugador, si es mayor, menor o igual que el correspondiente en la combinación secreta.

package main;

import java.util.Arrays;

import java.util.Random;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

Random rand = new Random();

System.out.println("Escribe la longitud de la clave");

int longitudCombi = sc.nextInt();

int []combi = new int [longitudCombi];

int [] combiUser = new int [longitudCombi];

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

combi[i] = rand.nextInt(1, 6);

}

System.out.println("La clave es " + Arrays.toString(combi));

System.out.println("Introduzca la clave secreta");

for (int digUser = 0; digUser < combiUser.length; digUser++) {

combiUser[digUser] = sc.nextInt();

}

boolean esCorrecto = false;

for (int dig = 0; dig < combi.length; dig++) {

if (combiUser[dig] > combi[dig]) {

System.out.println("Este número se pasa de largo");

} else if (combiUser[dig] < combi[dig] ){

System.out.println("Este número es muy pequeño");

} else if (combiUser[dig] == combi[dig]) {

System.out.println("El número está bien");

if (Arrays.equals(combi, combiUser)) {

System.out.println("Abriendo caja fuerte...");

}

}

}

}

}

11- Calcula la letra de un DNI, pediremos el DNI por teclado y nos devolverá el DNI completo. Para calcular la letra, cogeremos el resto de dividir nuestro DNI entre 23, el resultado debe estar entre 0 y 22. Haz un método donde según el resultado de la anterior formula busque en un array de caracteres la posición que corresponda a la letra. Esta es la tabla de caracteres: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 T R W A G M Y F P D X B N J Z S Q V H L C K E

**package** main;

**import** java.util.Scanner;

**public** **class** Main {

**public** **static** **void** main(String[] args) {

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Introduce el dni sin letra");

String dniSinLetra = sc.nextLine();

**int** dniNum = Integer.*parseInt*(dniSinLetra);

**int** resto = dniNum % 23;

**int** indBus = 0;

**for** (**int** comprob = 0; comprob < dniSinLetra.length(); comprob++) {

**if** (dniSinLetra.length() != 8) {

System.***out***.println("Formato no válido, vuelve a introducir el número");

dniSinLetra = sc.nextLine();

}

}

**char** [] letra = {'T', 'R', 'W', 'A', 'G', 'M', 'Y', 'F', 'P', 'D', 'X', 'B', 'N', 'J', 'Z', 'S', 'Q', 'V', 'H', 'L', 'C', 'K', 'E'};

**char** dniLetra = letra[resto];

System.***out***.println("El dni entero es " + dniSinLetra + dniLetra);

}

}

12- Generar un array de longitud n con números aleatorios no repetidos entre sí.