



Nombre: Morales Del Real Nairda Tonantzin	
Carrera: TSU en Tecnologías de la Información, Área Desarrollo de Software Multiplataforma.	Grupo: 4ATI
Asignatura: Estructuras de Datos Aplicadas	Unidad: Arreglos
Profesor: MGTI Jorge Luis De La Paz Ramos	

1. Diseña un programa con un arreglo de 20 elementos llamado pares y guarda los 20 primeros números pares en dicho arreglo, muestra en pantalla el contenido del arreglo.

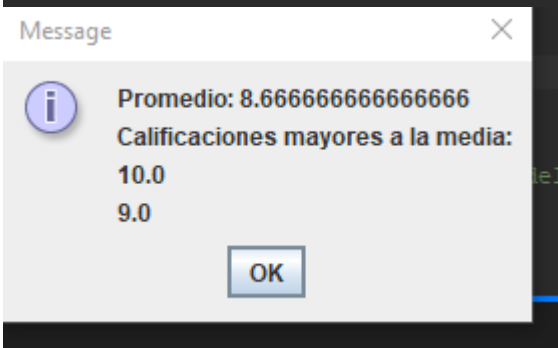
```
public class Ejercicio1 {  
  
    public static void main(String[] args) {  
        int[] par={2,4,6,8,10,12,14,16,18,20,22,24,26,28,30,32,34,36,38,40};  
        for(int i=0; i<par.length; i++){  
            System.out.print(par[i]+" , ");  
        }  
    }  
}
```

```
run:  
2 , 4 , 6 , 8 , 10 , 12 , 14 , 16 , 18 , 20 , 22 , 24 , 26 , 28 , 30 , 32 , 34 , 36 , 38 , 40
```

2. programa que llene por teclado la nota de los alumnos de una clase y calcula la nota media del grupo. también muestra los alumnos con notas superiores a la media. el número de alumnos se lee por teclado este programa utiliza un arreglo de elementos que contendrá las notas de los alumnos. El tamaño del arreglo será el número de alumnos de la clase por lo tanto primero se pedirá por teclado el número de alumnos y a continuación se creará el arreglo. se realizan 3 recorridos sobre el arreglo, el primera para asignar a cada elemento las notas introducidas por el teclado el segundo para sumarlasy calcular la media, y el tercero para mostrar los alumnos con notas superiores.



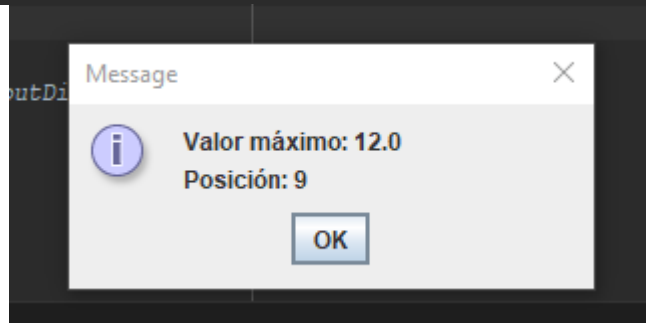
```
public class Arreglo2 {  
  
    /**  
     * @param args the command line arguments  
     */  
    public static void main(String[] args) {  
        int al = Integer.parseInt(JOptionPane.showInputDialog("Ingrese la cantidad de estudiantes:"));  
        double[] calificaciones = new double[al];  
        for (int i = 0; i < al; i++) {  
            calificaciones[i] = Double.parseDouble(JOptionPane.showInputDialog("Ingrese la calificación del alumno " + i + ":"));  
        }  
        double suma = 0;  
        for (int i = 0; i < al; i++) {  
            suma += calificaciones[i];  
        }  
        double media = suma / al;  
        String resultado = "Promedio: " + media + "\nCalificaciones mayores a la media:\n";  
        for (int i = 0; i < al; i++) {  
            if (calificaciones[i] > media) {  
                resultado += calificaciones[i] + "\n";  
            }  
        }  
        JOptionPane.showMessageDialog(null, resultado);  
    }  
}
```



3. Crear un arreglo que solicite 10 numeros y de ellos saque el digito mayor y muestre tambien su posicion dentro del arreglo. No se pueden introducir numeros repetidos.

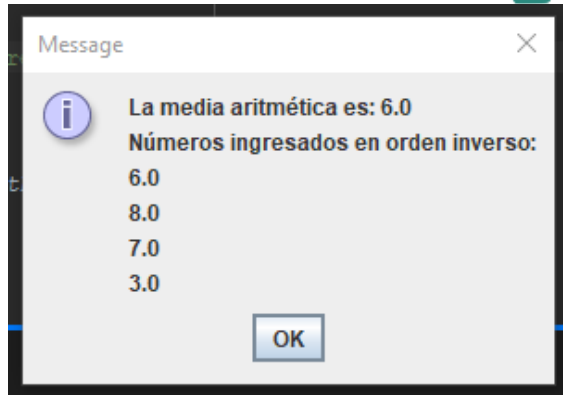
```
public static void main(String[] args) {
    double[] numeros = new double[10];
    int i = 0;
    while (i < 10) {
        double num = Double.parseDouble(JOptionPane.showInputDialog("Ingrese un número: "));
        int j;
        for (j = 0; j < i; j++) {
            if (numeros[j] == num) {
                break;
            }
        }
        if (j == i) {
            numeros[i] = num;
            i++;
        } else {
            JOptionPane.showMessageDialog(null, "El número ya ha sido ingresado");
        }
    }
    double max = numeros[0];
    int pos = 0;

    for (i = 0; i < numeros.length; i++) {
        if (numeros[i] > max) {
            max = numeros[i];
            pos = i;
        }
    }
    JOptionPane.showMessageDialog(null, "Valor máximo: " + max + "\nPosición: " + pos);
}
```



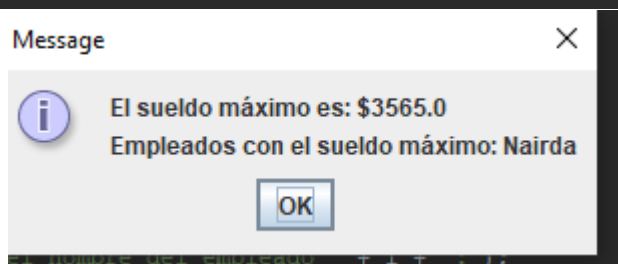
4. Realice un programa que calcule la media aritmética de N valores, además imprima los números ingresados en orden inverso.

```
public static void main(String[] args) {
    int N = Integer.parseInt(JOptionPane.showInputDialog("Ingrese la cantidad de valores:"));
    double[] valores = new double[N];
    double suma = 0;
    for (int i = 0; i < N; i++) {
        double num = Double.parseDouble(JOptionPane.showInputDialog("Ingrese el valor " + i + ":"));
        valores[i] = num;
        suma += num;
    }
    double media = suma / N;
    String resultado = "La media aritmética es: " + media + "\n";
    resultado += "Números ingresados en orden inverso:\n";
    for (int i = N - 1; i >= 0; i--) {
        resultado += valores[i] + "\n";
    }
    JOptionPane.showMessageDialog(null, resultado);
}
```



5. Programa que lea el nombre y el sueldo de 20 empleados y muestre el nombre y el sueldo del empleado que más gana. Si hay mas de un empleado que mostrar en el resultado, debe mostrar todos los que más ganan.

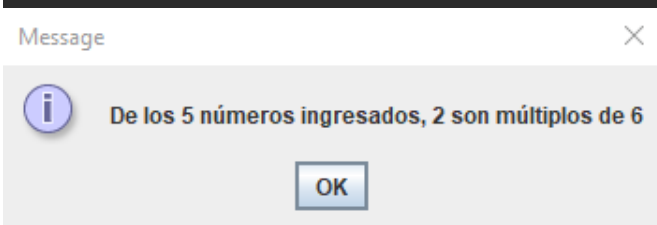
```
public static void main(String[] args) {
    int n = 20;
    String[] empleados = new String[n];
    double[] sueldos = new double[n];
    double maxs = 0;
    String ganadores = "";
    for (int i = 0; i < n; i++) {
        String nombre = JOptionPane.showInputDialog("Ingrese el nombre del empleado " + i + ":");
        double sueldo = Double.parseDouble(JOptionPane.showInputDialog("Ingrese el sueldo de " + nombre + ":"));
        empleados[i] = nombre;
        sueldos[i] = sueldo;
        if (sueldo > maxs) {
            maxs = sueldo;
            ganadores = nombre;
        } else if (sueldo == maxs) {
            ganadores += ", " + nombre;
        }
    }
    String resultado = "El sueldo máximo es: $" + maxs + "\n" +
        "Empleados con el sueldo máximo: " + ganadores;
    JOptionPane.showMessageDialog(null, resultado);
}
```



6. Dado 5 números y un divisor, determinar cuantos números múltiplos hay el divisor en los 5 números ingresados.

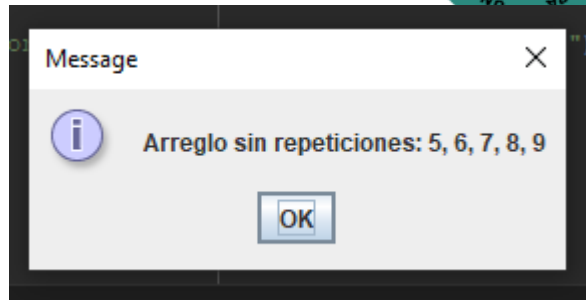


```
public static void main(String[] args) {  
    int[] numeros = new int[5];  
    int multiploCont = 0;  
    int divisor = Integer.parseInt(JOptionPane.showInputDialog("Ingrese el divisor:"));  
    for (int i = 0; i < 5; i++) {  
        int num = Integer.parseInt(JOptionPane.showInputDialog("Ingrese el número " + i + ":"));  
        numeros[i] = num;  
    }  
    for (int i = 0; i < numeros.length; i++) {  
        if (numeros[i] % divisor == 0) {  
            multiploCont++;  
        }  
    }  
    String resultado = "De los 5 números ingresados, " + multiploCont + " son múltiplos de " + divisor;  
    JOptionPane.showMessageDialog(null, resultado);  
}
```



7. Escriba un programa que reciba como entrada un arreglo unidimensional ordenado de enteros (posiblemente repetidos) y genere como salida una lista de los números enteros, pero sin repeticiones.

```
public static void main(String[] args) {  
    String entrada = JOptionPane.showInputDialog("Ingrese un arreglo ordenado de enteros separados por comas:");  
    String[] strArreglo = entrada.split(",");  
    int[] arreglo = new int[strArreglo.length];  
    for (int i = 0; i < strArreglo.length; i++) {  
        arreglo[i] = Integer.parseInt(strArreglo[i].trim());  
    }  
    int[] sinRepetidos = new int[arreglo.length];  
    int j = 0;  
    for (int i = 0; i < arreglo.length; i++) {  
        if (i == 0 || arreglo[i] != arreglo[i - 1]) {  
            sinRepetidos[j++] = arreglo[i];  
        }  
    }  
    String resultado = "Arreglo sin repeticiones: ";  
    for (int i = 0; i < j; i++) {  
        resultado += sinRepetidos[i];  
        if (i < j - 1) {  
            resultado += ", ";  
        }  
    }  
    JOptionPane.showMessageDialog(null, resultado);  
}
```



8. Lea 4 números y almacénelo en un arreglo llamado "A", y otros 4 números en un arreglo llamada "B", determine cuantos números de A se encuentran en "B".

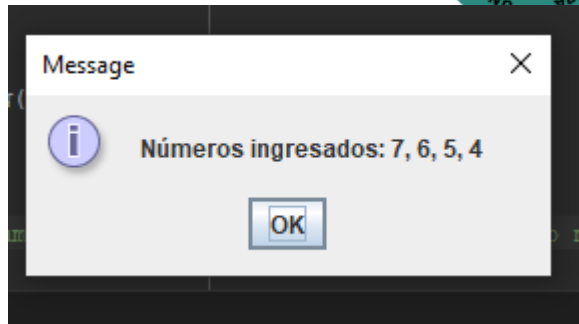
```
public static void main(String[] args) {
    int [] A = {1, 2, 3, 4};
    int [] B = {4, 2, 2, 8};
    int coincidencias = 0;

    for (int i = 0; i < A.length; i++) {
        for (int j = 0; j < B.length; j++) {
            if (A[i] == B[j]) {
                coincidencias++;
            }
        }
        System.out.println("Número " + A[i] + " se encuentra " + coincidencias + " veces en B.");
        coincidencias = 0;
    }
}
```

```
run:
Número 1 se encuentra 0 veces en B.
Número 2 se encuentra 2 veces en B.
Número 3 se encuentra 0 veces en B.
Número 4 se encuentra 1 veces en B.
```

9. Crear un programa que lea la cantidad de números deseada por el cliente y no permita introducir números repetidos.

```
public static void main(String[] args) {
    int cant = Integer.parseInt(JOptionPane.showInputDialog("Ingrese la cantidad de números:"));
    int[] nums = new int[cant];
    int numingres = 0;
    for (int i = 0; i < cant; i++) {
        int num;
        int rep;
        do {
            rep = 0;
            num = Integer.parseInt(JOptionPane.showInputDialog("Ingrese el número " + i + ":"));
            for (int j = 0; j < numingres; j++) {
                if (nums[j] == num) {
                    rep = 1;
                    JOptionPane.showMessageDialog(null, "El número ya ha sido ingresado. Ingrese otro número");
                    break;
                }
            }
        } while (rep == 1);
        nums[numingres++] = num;
    }
    String res = "Números ingresados: ";
    for (int i = 0; i < numingres; i++) {
        res += nums[i];
        if (i < numingres - 1) {
            res += ", ";
        }
    }
    JOptionPane.showMessageDialog(null, res);
}
```



10. Desarrollar un programa que rellene un arreglo con los 100 primeros números enteros y los muestre en pantalla en orden ascendente. Menor a mayor.

```
public static void main(String[] args) {  
    int[] numeros = new int[101];  
    for (int i = 0; i < 101; i++) {  
        numeros[i] = i;  
    }  
    System.out.println("Los primeros 100 números son:");  
    for (int i = 0; i < numeros.length; i++) {  
        System.out.println(numeros[i]);  
    }  
}
```

```
Los primeros 100 números son:  
0  
1  
2  
3  
4  
5  
6  
7  
8  
9
```

11. Crear un programa que lea 100 números por teclado y los ordene descendientemente. Mayor a menor



```
package ejerciciol1;
import java.util.Arrays;
import javax.swing.JOptionPane;
public class Ejerciciol1 {
    public static void main(String[] args) {
        int [] numeros = new int [100];

        for(int i = 0; i < 100; i++){
            String num = JOptionPane.showInputDialog("Introduce el número " + i);
            numeros[i] = Integer.parseInt(num);
        }

        String mensaje = "de mayor a menor: ";
        for (int i = numeros.length - 1; i >= 0; i--) {
            mensaje += numeros[i] + " ";
        }

        JOptionPane.showMessageDialog(null, mensaje);
    }
}

: 99, 98, 97, 96, 95, 94, 93, 92, 91, 90,
```

12. Programa que llena un arreglo con los números impares en el rango del 1 al 100.

```
package ejerciciol2;
import java.util.Arrays;
import javax.swing.JOptionPane;
public class Ejerciciol2 {

    public static void main(String[] args) {
        int[] numeros = new int[100];

        for (int i = 0; i < 100; i++) {
            numeros[i] = 2*i+1;
        }

        String resultado = Arrays.toString(numeros);
        System.out.println( resultado);
    }
}

run:
[1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63, 65, 67, 69, 71, 73, 75, 77, 79, 81, 83, 85, 87, 89, 91, 93, 95, 97, 99]
```

13. desarrollar un programa que lea 10 números por teclado los almacenen en un arreglo y muestre la suma resta multiplicación y división de todos.



```
package ejercicio13;
import javax.swing.JOptionPane;
public class Ejercicio13 {

    public static void main(String[] args) {
        int [] numeros = new int [10];

        for(int i = 0; i < 10; i++){
            numeros[i] = Integer.parseInt(JOptionPane.showInputDialog("Ingrese un número "+i));
        }

        double suma = 0;
        double mult = 1;
        double resta = numeros[0];
        double div = numeros[0];

        for( int i = 0; i < numeros.length; i++){
            suma += numeros[i];
            mult += numeros[i];
            if( i > 0){
                resta -= numeros[i];
                div /= numeros[i];
            }
        }

        String mensaje="suma del arreglo: "+suma+"\n"+
            "resta del arreglo: "+resta+"\n"+
            "multiplicacion del arreglo: "+mult+"\n"+
            "division del arreglo: "+div;

        JOptionPane.showMessageDialog(null, mensaje);
    }
}
```

Message



suma del arreglo: 356.0
resta del arreglo: -348.0
multiplicacion del arreglo: 357.0
division del arreglo: 2.1426268983132773E-13

OK

14. Realizar un programa que encuentre los numeros impares del 1 al 100 y los ordene y muestre en pantalla decendentemente.

```
package ejercicio14;
import javax.swing.JOptionPane;
public class Ejercicio14 {

    public static void main(String[] args) {
        int[] numeros = new int[100];

        for (int i = 0; i < 100; i++) {
            numeros[i] = 2*i+1;
        }

        java.util.Arrays.sort(numeros);

        String mensaje = "";
        for (int i = 0; i < numeros.length; i++) {
            mensaje += numeros[i] + " ";
        }

        JOptionPane.showMessageDialog(null, mensaje);
    }
}
```

run:

1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31 33 35 37 39 41 43 45 47 49 51 53 55 57 59 61 63 65 67 69 71 73 75 77 79 81 83 85 87 89 91 93 95 97 99



15. Realizar un programa que genere 100 numeros al azar y los ordene decendentemente

```
package ejercicio15;
import javax.swing.JOptionPane;
import java.util.Arrays;
public class Ejercicio15 {

    public static void main(String[] args) {
        int [] numeros = new int [100];

        for(int i = 0; i < 100; i++){
            double num = Math.floor(Math.random() * 100);
            numeros[i] = (int) num;
        }

        Arrays.sort(numeros);
        String resultado = Arrays.toString(numeros);
        System.out.println(resultado);
    }
}
```

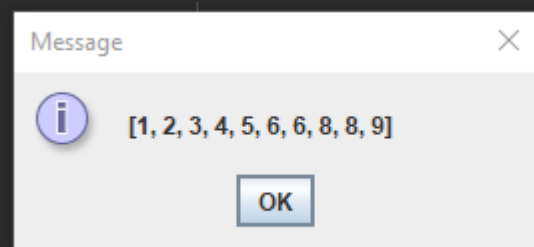
run:
[0, 1, 1, 1, 2, 2, 2, 5, 5, 5, 6, 6, 6, 6, 11, 12, 13, 13, 15, 15, 16, 20, 21, 22, 22, 24, 24, 26, 26, 28, 29, 29, 29, 31, 31, 31, 31, 32, 36, 37, 39, 40, 41, 42, 47, 48, 49, 50, 50, 51]

16. Realice un programa que tome 10 numeros y los ordene acendentemente

```
package ejercicio16;
import javax.swing.JOptionPane;
import java.util.Arrays;
public class Ejercicio16 {

    public static void main(String[] args) {
        int [] numeros = {5, 9, 6, 2, 8, 1, 3, 8, 6, 4};

        Arrays.sort(numeros);
        String resultado = Arrays.toString(numeros);
        JOptionPane.showMessageDialog(null, resultado);
    }
}
```





17. Desarrollar un programa que lea 10 números por teclado 5 para un arreglo y 5 para otro arreglo distinto mostrar los 10 números en pantalla mediante un solo arreglo

```
package ejercicio17;
import javax.swing.JOptionPane;
public class Ejercicio17 {

    public static void main(String[] args) {

        int [] arr1 = new int [5];
        int [] arr2 = new int [5];

        for(int i = 0; i < 5; i++) {
            String num1 = JOptionPane.showInputDialog("Ingrese un número para el arreglo 1: ");
            arr1[i] = Integer.parseInt(num1);
        }
        for(int i = 0; i < 5; i++) {
            String num2 = JOptionPane.showInputDialog("Ingrese un número para el arreglo 2: ");
            arr2[i] = Integer.parseInt(num2);
        }
        StringBuilder resultado = new StringBuilder("Arreglo combinado: ");
        for (int i = 0; i < 5; i++) {
            resultado.append(arr1[i]).append(", ");
            resultado.append(arr2[i]).append(", ");
        }
        System.out.println(resultado.toString());
    }
}

run:
Arreglo combinado: 6, 9, 7, 5, 8, 3, 5, 2, 3, 8,
```

18. Desarrolle un programa que lea 5 números por teclado los copie a otro arreglo multiplicado por 2 y muestre todos los datos ordenados ascendentemente usando un 3er arreglo.

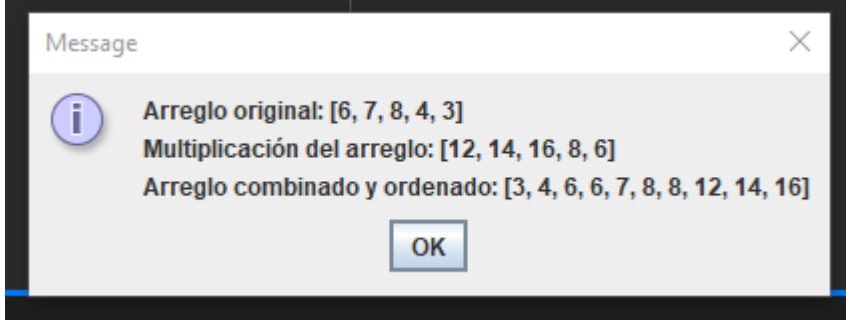
```
package ejercicio18;
import javax.swing.JOptionPane;
import java.util.Arrays;
public class Ejercicio18 {

    public static void main(String[] args) {
        int [] n = new int [5];
        for (int i = 0; i < 5; i++) {
            String num = JOptionPane.showInputDialog("Ingrese el número "+i);
            n[i] = Integer.parseInt(num);
        }

        int [] num_mults = new int [5];
        for (int i = 0; i < n.length; i++) {
            num_mults[i] = n[i] * 2;
        }

        int[] arr_comp = new int[n.length + num_mults.length];
        for (int i = 0; i < n.length; i++) {
            arr_comp[i] = n[i];
        }
        for (int i = 0; i < num_mults.length; i++) {
            arr_comp[n.length + i] = num_mults[i];
        }
        Arrays.sort(arr_comp);

        String or = Arrays.toString(n);
        String mult = Arrays.toString(num_mults);
        String mensaje = "Arreglo original: " +or+ "\n" +
            "Multiplicación del arreglo: " +mult+ "\n" +
            "Arreglo combinado y ordenado: "+Arrays.toString(arr_comp);
    }
}
```



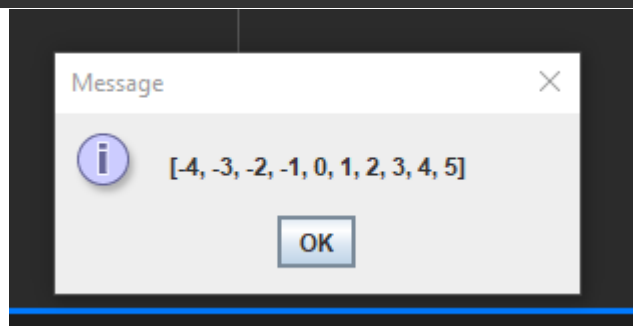
19. Desarrollar un programa que mediante un arreglo almacene números tanto positivos como negativos y los muestre ordenados ascendentemente.

```
package ejercicio19;
import javax.swing.JOptionPane;
import java.util.Arrays;
public class Ejercicio19 {

    public static void main(String[] args) {
        int [] n = {-1, 5, 2, -3, 1, 0, -4, 3, -2, 4};

        Arrays.sort(n);

        String m = Arrays.toString(n);
        JOptionPane.showMessageDialog(null, m);
    }
}
```



20. Desarrolle un programa que rellene un arreglo con 20 números y luego busque un número en concreto y devuelva la posición del mismo.

```
package ejercicio20;
import java.util.Arrays;
import javax.swing.JOptionPane;
public class Ejercicio20 {

    public static void main(String[] args) {
        int [] numeros = new int [20];

        for (int i = 0; i < 20; i++) {
            String num = JOptionPane.showInputDialog("Ingrese un número: "+i);
            numeros[i] = Integer.parseInt(num);
        }
        String m = "Arreglo ingresado: " + Arrays.toString(numeros) + "\n";

        String numBuscar = JOptionPane.showInputDialog("Ingrese un número para buscar: ");
        int numb = Integer.parseInt(numBuscar);

        int posicion = -1;
        for (int i = 0; i < numeros.length; i++) {
            if (numeros[i] == numb) {
                posicion = i;
                break;
            }
        }
        if (posicion == -1) {
            m += ("El número no se encuentra");
        } else {
            m += ("Número buscado: "+numb+"\nPosición: "+posicion);
        }
    }
}
```

run:

```
Arreglo ingresado: [6, 7, 5, 4, 6, 8, 4, 3, 5, 33, 2, 5, 67, 7, 89, 6, 4, 8, 4, 3]
Número buscado: 2
Posición: 10
```

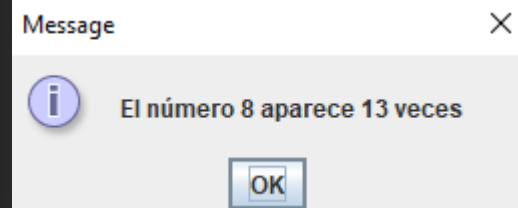
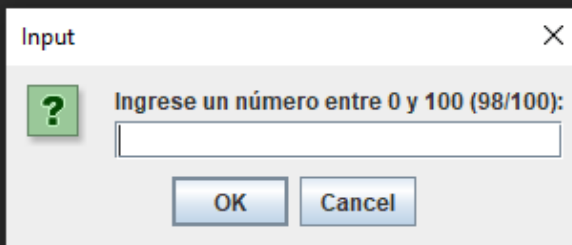


21. Construya un programa que realice las siguientes acciones. Considere un arreglo ordenado en forma ascendente

- Genere un arreglo de 100 números aleatorios entre 0 y 100
- Dado un número ingresado por el usuario, de como resultado la posición de la primera ocurrencia.
- Dado un número ingresado ver cuantas veces se repite en el arreglo.
- Muestre el arreglo las veces que el usuario lo requiera.

```
public static void main(String[] args) {  
    int [] arreglo = new int [100];  
  
    for (int i = 0; i < 100; i++) {  
        arreglo[i]=Integer.parseInt(JOptionPane.showInputDialog("Ingrese un número entre 0 y 100 (" + i + "/" + 100 + "):"));  
    }  
  
    for (int i = 0; i < arreglo.length; i++) {  
        for (int j = i + 1; j < arreglo.length; j++) {  
            if (arreglo[i] > arreglo[j]) {  
                int temp = arreglo[i];  
                arreglo[i] = arreglo[j];  
                arreglo[j] = temp;  
            }  
        }  
    }  
  
    int opcion;  
    do {  
        opcion = Integer.parseInt(JOptionPane.showInputDialog("Seleccione una opción:\n" +  
            "1. Buscar primera ocurrencia\n" +  
            "2. Contar repeticiones\n" +  
            "3. Mostrar el arreglo\n" +  
            "4. Salir"));  
  
        if (opcion == 1) {  
            int numBuscar = Integer.parseInt(JOptionPane.showInputDialog("Número a buscar:"));  
            var encontrado = false;  
        }  
    } while (opcion != 4);  
}
```

```
for (int i = 0; i < arreglo.length; i++) {  
    if (arreglo[i] == numBuscar) {  
        JOptionPane.showMessageDialog(null, "Primera ocurrencia en posición: " + i);  
        encontrado = true;  
        break;  
    }  
}  
if (!encontrado) {  
    JOptionPane.showMessageDialog(null, "Número no encontrado");  
}  
  
} else if (opcion == 2) {  
    int numContar = Integer.parseInt(JOptionPane.showInputDialog("Número a contar: "));  
    var contador = 0;  
  
    for (int i = 0; i < arreglo.length; i++) {  
        if (arreglo[i] == numContar) {  
            contador++;  
        }  
    }  
    JOptionPane.showMessageDialog(null, "El número " + numContar + " aparece " + contador + " veces");  
  
} else if (opcion == 3) {  
    for(int i = 0; i < arreglo.length; i++){  
        System.out.println("Arreglo: " + arreglo[i]);  
    }  
}  
} else if (opcion != 4) {  
    JOptionPane.showMessageDialog(null, "Opción no válida");  
}
```



22. Una compañía distribuye N productos a distintos comercios de la ciudad. Para ello almacena en un arreglo toda la información relacionada con su mercancía:

- Clave
- Descripción
- Existencia
- Mínimo para mantener de existencia.
- Precio Unitario.



```
public class Ejercicio22 {  
    public static void main(String[] args) {  
        int n = Integer.parseInt(JOptionPane.showInputDialog("Ingrese la cantidad de productos a registrar:"));  
  
        String[] claves = new String[n];  
        String[] descripciones = new String[n];  
        int[] existencias = new int[n];  
        int[] minimos = new int[n];  
        double[] preciosUnitarios = new double[n];  
  
        for (int i = 0; i < n; i++) {  
            claves[i] = JOptionPane.showInputDialog("Ingrese la clave del producto " + i + ":");  
            descripciones[i] = JOptionPane.showInputDialog("Ingrese la descripción del producto " + i + ":");  
            existencias[i] = Integer.parseInt(JOptionPane.showInputDialog("Ingrese la existencia del producto " + i + ":"));  
            minimos[i] = Integer.parseInt(JOptionPane.showInputDialog("Ingrese el mínimo para mantener existencia del producto " + i + ":"));  
            preciosUnitarios[i] = Double.parseDouble(JOptionPane.showInputDialog("Ingrese el precio unitario del producto " + i + ":"));  
        }  
  
        for (int i = 0; i < n; i++) {  
            JOptionPane.showMessageDialog(null, "Producto " + i + ":\n" +  
                "Clave: " + claves[i] + "\n" +  
                "Descripción: " + descripciones[i] + "\n" +  
                "Existencia: " + existencias[i] + "\n" +  
                "Mínimo: " + minimos[i] + "\n" +  
                "Precio Unitario: $" + String.format("%.2f", preciosUnitarios[i]));  
        }  
    }  
}
```

Message



Producto 0:

Clave: 101

Descripción: Camisa

Existencia: 45

Mínimo: 10

Precio Unitario: \$150.00

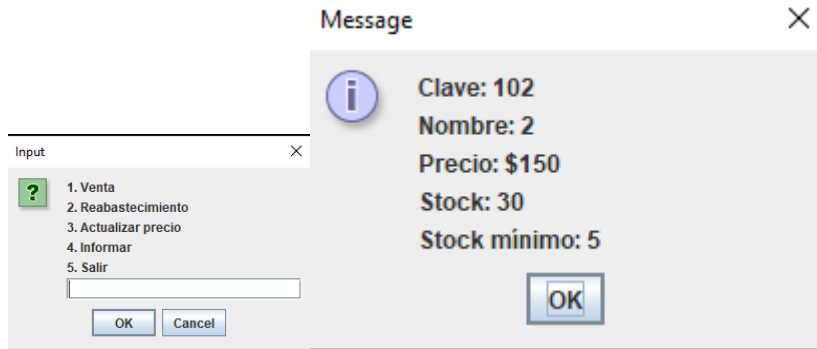


23. Escriba un programa que efectúe las siguientes operaciones:

- Venta de un producto: se deben actualizar los campos que correspondan y verificar que la nueva existencia no esté por debajo del mínimo. (Datos: clave, cantidad vendida)
- Reabastecimiento de un producto: se deben actualizar los campos que correspondan. (Datos: clave, cantidad comprada.)
- Actualizar el precio de un producto. (Datos: clave, porcentaje de aumento.)
- Informar sobre un productor: se deben proporcionar todos los datos relacionados con un producto. (Dato: clave)



```
public static void main(String[] args) {  
    int[] producto = {1,2,3};  
    int[] clave = {101,102,103};  
    int[] precio = {100,150,200};  
    int[] stock = {50,30,20};  
    int [] stockMin={10,5,8};  
  
    int opcion;  
    do {  
        opcion = Integer.parseInt(JOptionPane.showInputDialog("1. Venta\n2. Reabastecimiento\n3. Actualizar precio\n4. In  
  
        if (opcion == 1) {  
  
            int cla = Integer.parseInt(JOptionPane.showInputDialog("Ingrese la clave del producto:"));  
            int cantidad = Integer.parseInt(JOptionPane.showInputDialog("Ingrese la cantidad a vender:"));  
  
            for (int i = 0; i < producto.length; i++) {  
                if (clave[i] == cla) {  
                    if (stock[i] - cantidad >= stockMin[i]) {  
                        stock[i] -= cantidad;  
                        JOptionPane.showMessageDialog(null,"Venta realizada. Stock restante: " + stock[i]);  
                    } else {  
                        JOptionPane.showMessageDialog(null,"No se puede realizar la venta. Stock insuficiente.");  
                    }  
                }  
            }  
        } else if (opcion == 2) {  
  
            int cla = Integer.parseInt(JOptionPane.showInputDialog("Ingrese la clave del producto:"));  
            int cantidad = Integer.parseInt(JOptionPane.showInputDialog("Ingrese la cantidad a reabastecer:"));  
            int cla = Integer.parseInt(JOptionPane.showInputDialog("Ingrese la clave del producto:"));  
            int cantidad = Integer.parseInt(JOptionPane.showInputDialog("Ingrese la cantidad a reabastecer:"));  
  
            for (int i = 0; i < clave.length; i++) {  
                if (clave[i] == cla) {  
                    stock[i] += cantidad;  
                    JOptionPane.showMessageDialog(null,"Reabastecimiento realizado <br>Nuevo stock: " + stock[i]);  
                }  
            }  
        } else if (opcion == 3) {  
  
            int cla = Integer.parseInt(JOptionPane.showInputDialog("Clave del producto:"));  
            int porcentaje = Integer.parseInt(JOptionPane.showInputDialog("Porcentaje de aumento:"));  
  
            for (int i = 0; i < clave.length; i++) {  
                if (clave[i] == cla) {  
                    precio[i] += precio[i] * (porcentaje / 100);  
                    JOptionPane.showMessageDialog(null,"Nuevo precio: $" +precio[i]);  
                }  
            }  
        } else if (opcion == 4) {  
  
            int cla = Integer.parseInt(JOptionPane.showInputDialog("Clave del producto:"));  
  
            for (int i = 0; i < clave.length; i++) {  
                if (clave[i] == cla) {  
                    JOptionPane.showMessageDialog(null,  
                        "Clave: " + clave[i] +  
                        "\nNombre: " + producto[i] +  
                        "\nPrecio: $" + precio[i] +  
                        "\nStock: " + stock[i] +
```



24. Al momento de su ingreso al hospital, a un paciente se le solicitan los siguientes datos:

- Nombre
- Edad
- Sexo
- Domicilio:
 - Calle
 - Número
 - Ciudad
- Teléfono
- Seguro (este campo tendrá el valor VERDADERO si el paciente tiene seguro médico y FALSO en otro caso).

Escriba un programa que pueda llevar a cabo las siguientes operaciones:

- a) Listar los nombres de todos los pacientes hospitalizados.
- b) Obtener el porcentaje de pacientes hospitalizados en las siguientes categorías (dadas por la edad):
 - a. Niños: hasta 13 años.
 - b. Jóvenes: mayores de 13 años y menores de 30
 - c. Adultos: mayores de 30 años.
- c) Obtener el porcentaje de hombres y de mujeres hospitalizados.
- d) Dado el nombre de un paciente, listar todos los datos relacionados con dicho paciente.

```
public class Ejercicio24 {  
  
    /**  
     * @param args the command line arguments  
     */  
    public static void main(String[] args) {  
        int n = Integer.parseInt(JOptionPane.showInputDialog("Ingrese la cantidad de pacientes a registrar:"));  
  
        String[] nombres = new String[n];  
        int[] edades = new int[n];  
        String[] sexos = new String[n];  
        String[] calles = new String[n];  
        String[] numeros = new String[n];  
        String[] ciudades = new String[n];  
        String[] telefonos = new String[n];  
        String[] seguros = new String[n];  
  
        for (int i = 0; i < n; i++) {  
            nombres[i] = JOptionPane.showInputDialog("Ingrese el nombre del paciente " + i + ":");  
            edades[i] = Integer.parseInt(JOptionPane.showInputDialog("Ingrese la edad de " + nombres[i] + ":"));  
            sexos[i] = JOptionPane.showInputDialog("Ingrese el sexo de " + nombres[i] + " (M/F):");  
            calles[i] = JOptionPane.showInputDialog("Ingrese la calle de domicilio de " + nombres[i] + ":");  
            numeros[i] = JOptionPane.showInputDialog("Ingrese el número de domicilio de " + nombres[i] + ":");  
            ciudades[i] = JOptionPane.showInputDialog("Ingrese la ciudad de " + nombres[i] + ":");  
            telefonos[i] = JOptionPane.showInputDialog("Ingrese el teléfono de " + nombres[i] + ":");  
            String respuestaSeguro = JOptionPane.showInputDialog("¿Tiene seguro? (si/no):").toLowerCase();  
            seguros[i] = respuestaSeguro.equals("si") ? "Si" : "No";  
        }  
    }  
}
```



```
do {
    opcion = JOptionPane.showInputDialog("Seleccione una opción:\n" +
        "a. Listar todos los nombres de pacientes\n" +
        "b. Ver porcentaje por categoría de edad\n" +
        "c. Ver porcentaje de hombres y mujeres\n" +
        "d. Buscar datos de un paciente\n" +
        "e. Salir");

    if (opcion.equals("a")) {
        StringBuilder listaNombres = new StringBuilder("Pacientes registrados:\n");
        for (int i = 0; i < nombres.length; i++) {
            listaNombres.append((i + 1)).append(" ").append(nombres[i]).append("\n");
        }
        JOptionPane.showMessageDialog(null, listaNombres.toString());
    } else if (opcion.equals("b")) {
        int ninos = 0, jovenes = 0, adultos = 0;
        for (int edad : edades) {
            if (edad <= 13) {
                ninos++;
            } else if (edad > 13 && edad < 30) {
                jovenes++;
            } else {
                adultos++;
            }
        }
        JOptionPane.showMessageDialog(null, "Porcentaje de Niños: " + ((ninos / (double) n) * 100) + "%\n" +
            "Porcentaje de Jóvenes: " + ((jovenes / (double) n) * 100) + "%\n" +
            "Porcentaje de Adultos: " + ((adultos / (double) n) * 100) + "%");
    } else if (opcion.equals("c")) {
        int hombres = 0, mujeres = 0;
        for (String sexo : sexos) {
            if (sexo.equalsIgnoreCase("M")) {
                hombres++;
            } else if (sexo.equalsIgnoreCase("F")) {
                mujeres++;
            }
        }
        JOptionPane.showMessageDialog(null, "Porcentaje de Hombres: " + ((hombres / (double) n) * 100) + "%\n" +
            "Porcentaje de Mujeres: " + ((mujeres / (double) n) * 100) + "%");
    } else if (opcion.equals("d")) {
        String nombreBuscar = JOptionPane.showInputDialog("Ingrese el nombre del paciente que desea buscar:");
        boolean encontrado = false;
        for (int i = 0; i < nombres.length; i++) {
            if (nombres[i].equalsIgnoreCase(nombreBuscar)) {
                JOptionPane.showMessageDialog(null, "Datos de " + nombres[i] + ":\n" +
                    "Edad: " + edades[i] + "\n" +
                    "Sexo: " + sexos[i] + "\n" +
                    "Domicilio: Calle " + calles[i] + ", " + numeros[i] + ", " + ciudades[i] + "\n" +
                    "Teléfono: " + telefonos[i] + "\n" +
                    "Seguro: " + seguros[i]);
                encontrado = true;
                break;
            }
        }
        if (!encontrado) {
            JOptionPane.showMessageDialog(null, "Paciente no encontrado.");
        }
    } else if (opcion.equals("e")) {
        JOptionPane.showMessageDialog(null, "Opción no válida. Intente de nuevo.");
    }
}
```

Message



Datos de Nairda:

Edad: 19

Sexo: F

Domicilio: Calle Libertad, 8, Escuinapa

Teléfono: 6959513289

Seguro: Sí





25. Supongamos que en una elección hubo 12 candidatos (con identificadores 1,2,3, ..., 12). Por otra parte, los votos para cada candidato se teclean de manera desorganizada como se muestra a continuación:

1 5 7 5 1 12 10 7 1 7 5 8 1 5 -1(número negativo para detener la captura de votos)

Construya un programa que pueda proporcionar la siguiente información:

- El número de votos de cada candidato al final de la elección.
- El candidato ganador, el número de votos que obtuvo y el porcentaje correspondiente del total de la elección. Suponemos que el candidato ganador no empató en número de votos con otro candidato.

```
public class Ejercicio25 {  
  
    public static void main(String[] args) {  
        int[] votos = new int[12];  
  
        int voto = Integer.parseInt(JOptionPane.showInputDialog("Ingrese el número del candidato (1-12) para registrar un voto, o un número negativo para terminar:"));  
        while (voto >= 0) {  
            if (voto >= 1 && voto <= 12) {  
                votos[voto - 1]++;  
            } else {  
                JOptionPane.showMessageDialog(null, "Número inválido, por favor ingrese un candidato del 1 al 12.");  
            }  
            voto = Integer.parseInt(JOptionPane.showInputDialog("Ingrese el número del candidato (1-12) para registrar un voto, o un número negativo para terminar:"));  
        }  
  
        StringBuilder resultado = new StringBuilder("Resultados de la elección:\n");  
        int totalVotos = 0;  
        for (int i = 0; i < votos.length; i++) {  
            resultado.append("Candidato ").append(i + 1).append(": ").append(votos[i]).append(" votos\n");  
            totalVotos += votos[i];  
        }  
        JOptionPane.showMessageDialog(null, resultado.toString());  
  
        int maxVotos = votos[0];  
        int candidatoGanador = 1;  
        for (int i = 1; i < votos.length; i++) {  
            if (votos[i] > maxVotos) {  
                maxVotos = votos[i];  
                candidatoGanador = i + 1;  
            }  
        }  
  
        int maxVotos = votos[0];  
        int candidatoGanador = 1;  
        for (int i = 1; i < votos.length; i++) {  
            if (votos[i] > maxVotos) {  
                maxVotos = votos[i];  
                candidatoGanador = i + 1;  
            }  
        }  
  
        double porcentajeGanador = (maxVotos / (double) totalVotos) * 100;  
        JOptionPane.showMessageDialog(null, "Candidato ganador: " + candidatoGanador + "\n" +  
            "Votos obtenidos: " + maxVotos + "\n" +  
            "Porcentaje del total de votos: " + String.format("%.2f", porcentajeGanador) + "%");  
    }  
}
```



Message



Resultados de la elección:

Candidato 1: 1 votos
Candidato 2: 0 votos
Candidato 3: 0 votos
Candidato 4: 1 votos
Candidato 5: 1 votos
Candidato 6: 1 votos
Candidato 7: 1 votos
Candidato 8: 10 votos
Candidato 9: 1 votos
Candidato 10: 1 votos
Candidato 11: 3 votos
Candidato 12: 0 votos



Message



Candidato ganador: 8
Votos obtenidos: 10
Porcentaje del total de votos: 50.00%



26. Almacene en un arreglo las temperaturas de un día (0-23 horas). Calcule su media e imprimir la temperatura más alta y la más baja; asimismo imprimir la hora respectiva. Por ejemplo, la temperatura media fue de 21.5 grados Celsius, la más alta de 29 grados Celsius a las 14 horas y la más baja de 9.4 grados Celsius a las 3 horas.

```
public static void main(String[] args) {  
  
    double [] temperaturas = new double [24];  
  
    for (var i = 0; i < 24; i++) {  
        temperaturas[i] = Double.parseDouble(JOptionPane.showInputDialog("Ingrese la temperatura para la hora " + i + " :"));  
    }  
  
    var sumaTemperaturas = 0;  
    for (var i = 0; i < temperaturas.length; i++) {  
        sumaTemperaturas += temperaturas[i];  
    }  
    var temperaturaMedia = sumaTemperaturas / temperaturas.length;  
  
    var tempMax = temperaturas[0];  
    var tempMin = temperaturas[0];  
    var horaMax = 0;  
    var horaMin = 0;  
  
    for (var i = 1; i < temperaturas.length; i++) {  
        if (temperaturas[i] > tempMax) {  
            tempMax = temperaturas[i];  
            horaMax = i;  
        }  
        if (temperaturas[i] < tempMin) {  
            tempMin = temperaturas[i];  
            horaMin = i;  
        }  
    }  
  
    System.out.println("La temperatura media fue de " + temperaturaMedia + " grados Celsius"\n"La temperatura más alta fue de " + tempMax +  
        " grados Celsius a las " + horaMax + " horas"\n"La temperatura más baja fue de " + tempMin + " grados Celsius a las " + horaMin + " horas");  
  
    run:  
    La temperatura media fue de 30 grados Celsius  
    La temperatura más alta fue de 37.0 grados Celsius a las 3 horas  
    La temperatura más baja fue de 24.0 grados Celsius a las 23 horas
```



27. Almacene en un arreglo la temperatura de cada día de una determinada semana y que realice lo siguiente:
- La temperatura promedio.
 - Un arreglo que contenga las diferencias de cada temperatura con respecto al promedio.
 - La menor temperatura y el número de día en que ocurrió.
 - La mayor temperatura y el número de día en que ocurrió.

```
public class Ejercicio27 {  
  
    public static void main(String[] args) {  
        int[] semana = {15, 20, 18, 19, 22, 21, 17};  
        double promedio = 0;  
        double[] diferencias = new double[semana.length];  
        int minima = semana[0];  
        int minimoDia = 1;  
        int maxima = semana[0];  
        int maximoDia = 1;  
  
        for (int i = 0; i < semana.length; i++) {  
            promedio += semana[i];  
            if (semana[i] < minima) {  
                minima = semana[i];  
                minimoDia = i + 1;  
            }  
            if (semana[i] > maxima) {  
                maxima = semana[i];  
                maximoDia = i + 1;  
            }  
        }  
        promedio /= semana.length;  
  
        for (int i = 0; i < semana.length; i++) {  
            diferencias[i] = semana[i] - promedio;  
        }  
  
        StringBuilder resultado = new StringBuilder();  
        resultado.append("Temperatura promedio: ").append(promedio).append("\n");  
        resultado.append("Diferencias con respecto al promedio:\n");  
        for (int i = 0; i < diferencias.length; i++) {  
            resultado.append("Día ").append(i + 1).append(": ").append(diferencias[i]).append("\n");  
        }  
        resultado.append("Menor temperatura: ").append(minima).append(" en el día ").append(minimoDia).append("\n");  
        resultado.append("Mayor temperatura: ").append(maxima).append(" en el día ").append(maximoDia).append("\n");  
  
        JOptionPane.showMessageDialog(null, resultado.toString(), "Resultados de la Semana", JOptionPane.INFORMATION_MESSAGE);  
    }  
}
```



Resultados de la Semana



Temperatura promedio: 18.857142857142858

Diferencias con respecto al promedio:

Día 1: -3.8571428571428577

Día 2: 1.1428571428571423

Día 3: -0.8571428571428577

Día 4: 0.14285714285714235

Día 5: 3.1428571428571423

Día 6: 2.1428571428571423

Día 7: -1.8571428571428577

Menor temperatura: 15 en el día 1

Mayor temperatura: 22 en el día 5

OK

28. Almacene en un arreglo, 20 elementos de tipo entero e imprimir:
- La suma de los elementos que ocupan posiciones pares.
 - El mayor de los elementos que ocupan posiciones impares.
 - La posición del mayor número par.

```
public class Ejercicio28 {  
  
    public static void main(String[] args) {  
        int[] num = {3, 17, 8, 26, 14, 22, 11, 34, 5, 39, 12, 7, 4, 15, 29, 30, 18, 13, 21, 10};  
        int sumPP = 0;  
        for (int i = 0; i < num.length; i += 2) {  
            sumPP += num[i];  
        }  
  
        int mPI = Integer.MIN_VALUE;  
        for (int i = 1; i < num.length; i += 2) {  
            if (num[i] > mPI) {  
                mPI = num[i];  
            }  
        }  
        int mayorPar = Integer.MIN_VALUE;  
        int pMP = -1;  
        for (int i = 0; i < num.length; i++) {  
            if (num[i] % 2 == 0 && num[i] > mayorPar) {  
                mayorPar = num[i];  
                pMP = i;  
            }  
        }  
        JOptionPane.showMessageDialog(null, "Suma de los números en posiciones pares: " + sumPP + "\n" +  
            "Mayor de los números en posiciones impares: " + mPI + "\n" +  
            "Posición del mayor número par: " + pMP);  
    }  
}
```

Message



Suma de los números en posiciones pares: 125

Mayor de los números en posiciones impares: 39

Posición del mayor número par: 7

OK



29. Almacene en un arreglo, n elementos (máximo 30) e imprimir la suma:
- de números pares.
 - de números impares.
 - total de los elementos del arreglo.

```
public class Ejercicio29 {  
    /**  
     * @param args the command line arguments  
     */  
    public static void main(String[] args) {  
        int [] numeros = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30};  
  
        var sumaPares = 0;  
        var sumaImpares = 0;  
        var total = 0;  
  
        for (var i = 0; i < numeros.length; i++) {  
            if (numeros[i] % 2 == 0) {  
                sumaPares += numeros[i];  
            } else {  
                sumaImpares += numeros[i];  
            }  
            total += numeros[i];  
        }  
  
        System.out.println("Suma de numeros pares: " + sumaPares );  
        System.out.println("Suma de numeros impares: " + sumaImpares );  
        System.out.println("Total de elementos del arreglo: " + total);  
    }  
}
```

Suma de numeros pares: 240
Suma de numeros impares: 225
Total de elementos del arreglo: 465

30. Almacene en una lista 15 números e imprimir cuántos son ceros, cuántos son negativos y cuántos positivos. Imprimir además la suma de los negativos y la suma de los positivos.



```
public class Ejercicio30 {  
  
    public static void main(String[] args) {  
        int[] num = {5, -3, 0, 12, -8, 7, 0, 10, -5, 0, 2, -1, 15, 0};  
        var c = 0;  
        var n = 0;  
        var p = 0;  
        var sumN = 0;  
        var sumP = 0;  
  
        for (var i = 0; i < num.length; i++) {  
            if (num[i] == 0) {  
                c++;  
            } else if (num[i] < 0) {  
                n++;  
                sumN += num[i];  
            } else {  
                p++;  
                sumP += num[i];  
            }  
        }  
  
        System.out.println("ceros: " + c );  
        System.out.println("negativos: " + n + ", Suma negativos " + sumN );  
        System.out.println("positivos: " + p + ", Suma positivos: " + sumP );  
    }  
}  
  
run:  
ceros: 4  
negativos: 4, Suma negativos -17  
positivos: 6, Suma positivos: 51
```

31. Inicialice un arreglo con 10 valores. Genere un arreglo a con los números negativos y otro arreglo b con los positivos o iguales a cero.



```
public static void main(String[] args) {  
    int [] numeros = {22, -15, 0, 18, -63, 0, 17, -19, 4, -82};  
  
    int [] neg = new int [numeros.length];  
    int [] pos = new int [numeros.length];  
  
    int negC = 0;  
    int posC = 0;  
  
    for (int i = 0; i < numeros.length; i++){  
        if(numeros[i]<0){  
            neg [negC] = numeros [i];  
            negC++;  
        }else{  
            pos[posC] = numeros [i];  
            posC++;  
        }  
    }  
}
```

String numerosT = Arrays.toString(numeros);
System.out.println("Arreglo original: " + numerosT);

String posT = Arrays.toString(Arrays.copyOf(pos, posC));
System.out.println("Positivos: "+posT);

String negT = Arrays.toString(Arrays.copyOf(neg, negC));
System.out.println("Negativos: "+negT);

}

run:

```
Arreglo original: [22, -15, 0, 18, -63, 0, 17, -19, 4, -82]  
Positivos: [22, 0, 18, 0, 17, 4]  
Negativos: [-15, -63, -19, -82]
```

32. Almacene en un arreglo n elementos de tipo real (máximo 25) e imprima los siguientes valores:
- Máximo.
 - Mínimo.
 - La media de los que estén en posiciones pares.
 - La varianza. (<http://www.disfrutalasmaticas.com/datos/desviacion-estandar.html>)



```
public static void main(String[] args) {  
    int n = Integer.parseInt(JOptionPane.showInputDialog("Ingrese la cantidad de elementos: "));  
  
    int [] numeros = new int [n];  
    int sumPar = 0;  
    int numsPar = 0;  
  
    for(int i = 0; i < numeros.length; i++){  
        numeros[i] = Integer.parseInt(JOptionPane.showInputDialog("Ingrese el número "+i));  
        if(i % 2 == 0){  
            numsPar++;  
            sumPar = sumPar + numeros[i];  
        }  
    }  
  
    double media = sumPar / numsPar;  
  
    double max = numeros [0];  
    double min = numeros [0];  
  
    for(int i = 0; i < numeros.length; i++){  
        if(numeros[i]>max){  
            max = numeros[i];  
        }if(numeros[i]<min){  
            min = numeros[i];  
        }  
    }  
  
    int suma = 0;  
    double sumaDif = 0;  
    for(int i = 0; i < numeros.length; i++){  
        suma = suma + numeros[i];  
        double prom = suma / numeros.length;  
        double dif = (numeros[i] - prom) * (numeros[i] - prom);  
    }  
}
```

```
run:  
Número máximo: 89.0  
Número mínimo: 23.0  
Media de los numeros es posiciones pares: 56.0  
Varianza: 1129.5
```

33. Genere aleatoriamente un arreglo de tamaño 20 con números entre 0 y 150 y genere otras tres listas con los siguientes criterios:
- Si los números están comprendidos entre 0 y 50 irán en la lista 1.
 - Si los números están comprendidos entre 51 y 100 irán en la lista 2.
 - Si los números son mayores a 101 irán en la lista 3.
- Al final imprimir las cuatro listas.



```
int [] numeros = new int [20];
int [] nums1 = new int[numeros.length];
int [] nums2 = new int[numeros.length];
int [] nums3 = new int[numeros.length];

for (int i = 0; i < 20; i++) {
    numeros[i] = (int)(Math.random() * 151);
}

Arrays.sort(numeros);

int nums1C = 0;
int nums2C = 0;
int nums3C = 0;

for (var i = 0; i < numeros.length; i++) {
    if (numeros[i] < 51) {
        nums1[nums1C] = numeros[i];
        nums1C++;
    } else if (numeros[i] < 101) {
        nums2[nums2C] = numeros[i];
        nums2C++;
    } else {
        nums3[nums3C] = numeros[i];
        nums3C++;
    }
}

String numerosT = Arrays.toString(numeros);
System.out.println("numeros"+numerosT);

String nums1T = Arrays.toString(Arrays.copyOf(nums1, nums1C));
```

```
run:
numeros[3, 4, 13, 25, 35, 37, 53, 58, 60, 66, 69, 72, 72, 82, 87, 96, 101, 111, 133, 142]
Números entre 0 y 50: [3, 4, 13, 25, 35, 37]
Números entre 51 y 100: [53, 58, 60, 66, 69, 72, 72, 82, 87, 96]
Números mayores a 101: [101, 111, 133, 142]
```

34. Almacene en un arreglo 10 números y cambie algún número por otro del mismo arreglo, realice el cambio y muestre el arreglo modificado.
35. Genere aleatoriamente un arreglo de tamaño 20 con números comprendidos entre 1 y 20 e imprimir:
- Suma de los elementos.
 - Promedio de los elementos.
 - Varianza y desviación estándar de los valores.
 - Elemento que más se repite (moda).



```
public class Programa35_arreglos {

    public static void main(String[] args) {
        int [] numeros = new int [20];

        for (int i = 0; i < numeros.length; i++){
            numeros[i] = (int)(Math.random() * 20)+1;
        }

        Arrays.sort(numeros);

        String numerosT = Arrays.toString(numeros);
        System.out.println("Arreglo original: " + numerosT + "\n");

        int suma = 0;
        for(int i = 0; i < numeros.length; i++){
            suma = suma + numeros[i];
        }
        double promedio = suma / numeros.length;

        int sumaVar = 0;
        double sumaDif = 0;
        for(int i = 0; i < numeros.length; i++){
            sumaVar = sumaVar + numeros[i];
            double prom = sumaVar / numeros.length;
            double dif = (numeros[i] - prom) * (numeros[i] - prom);
            sumaDif = sumaDif + dif;
        }
        double prom = sumaVar / numeros.length;
        double dif = (numeros[i] - prom) * (numeros[i] - prom);
        sumaDif = sumaDif + dif;
    }
    double varianza = sumaDif / numeros.length;
    double desvEst = Math.sqrt(varianza);

    int masRep = numeros[0];
    int maxRep = 1;
    int reps = 1;

    for (int i = 1; i < numeros.length; i++) {
        if (numeros[i] == numeros[i - 1]) {
            reps++;
        } else {
            reps = 1;
        }

        if (reps > maxRep) {
            maxRep = reps;
            masRep = numeros[i];
        }
    }

    System.out.println("Suma de los elementos: " + suma);
    System.out.println("Promedio de los elementos: " + promedio);
    System.out.println("Varianza de los elementos: " + varianza);
    System.out.println("Desviación estándar de los elementos: " + desvEst);
    System.out.println("Número más repetido: " + masRep);
}

}
```



```
run:
Arreglo original: [1, 1, 3, 3, 4, 6, 6, 7, 9, 10, 11, 11, 12, 13, 14, 15, 16, 16, 16, 20]

Suma de los elementos: 194
Promedio de los elementos: 9.0
Varianza de los elementos: 51.4
Desviación estándar de los elementos: 7.1693793315739685
Número más repetido: 16
```

36. Almacene en un arreglo de tamaño 10 números reales. Calcule el promedio e indique cuántos elementos del arreglo son mayores que el promedio y genere otro arreglo con los menores o iguales.

```
public static void main(String[] args) {
    int[] numeros = {17, 28, 33, 45, 59, 63, 71, 88, 90, 100};

    int suma = 0;
    for (var i = 0; i < numeros.length; i++) {
        suma += numeros[i];
    }
    double prom = suma / numeros.length;

    int [] nums = new int [numeros.length];
    int may = 0;
    int menC = 0;
    for (var i = 0; i < numeros.length; i++) {
        if (numeros[i] > prom) {
            may++;
        } else {
            nums[menC] = numeros[i];
            menC++;
        }
    }

    String numerosT = Arrays.toString(numeros);
    System.out.println("Arreglo original: " + numerosT);
    //JOptionPane.showMessageDialog(null, "Arreglo: "+numerosT);

    System.out.println("Promedio de todos los números: " + prom);
    //JOptionPane.showMessageDialog(null, "Promedio de todos los números: " + prom);
    System.out.println("Cantidad de números mayores al promedio: " + may);
    //JOptionPane.showMessageDialog(null, "Cantidad de números mayores al promedio: " + may);

    String numsM = Arrays.toString(Arrays.copyOf(nums, menC));
    //JOptionPane.showMessageDialog(null, "Números menores o iguales al promedio: "+numsM);

    run:
    Arreglo original: [17, 28, 33, 45, 59, 63, 71, 88, 90, 100]
    Promedio de todos los números: 59.0
    Cantidad de números mayores al promedio: 5
    Números menores al promedio: [17, 28, 33, 45, 59]
```

37. Almacene en un arreglo de n números enteros (máximo 30) y determine cuántos números terminan en dígito primo.



```
public static void main(String[] args) {
    int[] numeros = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20};

    int cont = 0;
    for (int i = 0; i < numeros.length; i++){
        int ultDig = numeros[i] % 10;
        if(ultDig % 2 == 0){
            cont++;
        }
    }

    String numerosT = Arrays.toString(numeros);
    System.out.println("Arreglo: "+numerosT);

    System.out.println("Cantidad de números con último dígito par: " +cont);
}

run:
Arreglo: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]
Cantidad de números con último dígito par: 10
```

38. Convierta un número entero decimal a su equivalente en representación binaria. (http://recursostic.educacion.es/secundaria/edad/4esotecnologia/quincena5/4q2_contenidos_2c.htm)

```
package programa38_arreglos;
import javax.swing.*;

public class Programa38_arreglos {

    public static void main(String[] args) {
        int num = Integer.parseInt(JOptionPane.showInputDialog("Ingrese un número: "));
        System.out.println("Número: "+num);

        String binario = "";

        while (num > 0){
            int residuo = num % 2;
            binario = residuo + binario;
            num = num / 2;
        }

        System.out.println("El número en binario es: "+binario);
    }
}

run:
Número: 123
El número en binario es: 1111011
```



39. Almacene en un arreglo n números enteros y determine ¿cuántas veces se repite cada uno de ellos?

Si $n=6$ y los elementos del arreglo son: 3,4,6,6,4,6, se imprimirá:

3=1

4=2

6=3

```
public static void main(String[] args) {  
    int n = Integer.parseInt(JOptionPane.showInputDialog("Ingrese el tamaño del arreglo: "));  
  
    int [] numeros = new int [n];  
  
    for (int i = 0; i < numeros.length; i++){  
        numeros[i] = Integer.parseInt(JOptionPane.showInputDialog("Ingrese el número "+i));  
    }  
  
    String numerosT = Arrays.toString(numeros);  
    System.out.println("Arreglo: "+numerosT);  
  
    int masRep = numeros[0];  
    int maxRep = 1;  
    int reps = 1;  
  
    for (int i = 1; i < numeros.length; i++) {  
        if (numeros[i] == numeros[i - 1]) {  
            reps++;  
        } else {  
            reps = 1;  
        }  
  
        if (reps > maxRep) {  
            maxRep = reps;  
            masRep = numeros[i];  
        }  
    }  
  
    System.out.println("Número más repetido: " + masRep);  
}
```

Herramienta Recortes

Herramienta ...

```
run:  
Arreglo: [8, 7, 8, 4]  
Número más repetido: 8
```

40. Almacene en un arreglo n números (n debe ser un número par, y como máximo 50), valide si el tamaño es par e invertir los elementos a la mitad del arreglo. Imprimir el vector original y el invertido.

Si $n=6$, $v=[1][2][3][4][5][6]$ $v(\text{invertido})=[3][2][1][6][5][4]$.



```
public static void main(String[] args) {
    int cant = Integer.parseInt(JOptionPane.showInputDialog("Ingrese la cantidad de elementos para el arreglo (debe ser un número par y menor a 50): "));

    int [] numeros = new int [cant];

    if (cant % 2 == 0 && cant < 50){
        for(int i = 0; i < cant; i++){
            numeros[i] = i;
        }
        String numerosT = Arrays.toString(numeros);
        System.out.println("Arreglo original: "+numerosT);

        int mitad = cant / 2;

        for (int i = 0; i < mitad / 2; i++) {
            int temp = numeros[i];
            numeros[i] = numeros[mitad - i - 1];
            numeros[mitad - i - 1] = temp;
        }

        int[] resultado = new int[cant];
        System.arraycopy(numeros, 0, resultado, 0, cant);

        String resultadoT = Arrays.toString(resultado);
        System.out.println("Arreglo invertido: " + resultadoT);
    }
}

run:
Arreglo original: [0, 1, 2, 3]
Arreglo invertido: [1, 0, 2, 3]
BUILD SUCCESSFUL (total time: 18 seconds)
```

41. Almacene en dos arreglos números reales a y b de 10 elementos cada uno, a partir de ellos genere un tercer arreglo con el siguiente criterio: sumar el primer elemento de a más el último elemento de b y luego el segundo elemento de a con el noveno elemento de b y así sucesivamente hasta llegar al décimo elemento de a más el primer elemento de b. Imprimir las sumas almacenadas en el arreglo c.



```
public static void main(String[] args) {  
    int[] a = {1, 6, 9, 7, 3, 4, 2, 5, 8, 10};  
    int[] b = {9, 7, 6, 18, 3, 4, 8, 5, 2, 1};  
    int[] c = new int[a.length];  
  
    for (int i = 0; i < a.length; i++) {  
        int sum = a[i] + b[b.length - 1 - i];  
        c[i] = sum;  
    }  
  
    System.out.print("a: ");  
    for (int value : a) {  
        System.out.print(value + " ");  
    }  
    System.out.println();  
  
    System.out.print("b: ");  
    for (int value : b) {  
        System.out.print(value + " ");  
    }  
    System.out.println();  
  
    System.out.print("c: ");  
    for (int value : c) {  
        System.out.print(value + " ");  
    }  
    System.out.println();  
}
```

run:

```
a: 1 6 9 7 3 4 2 5 8 10  
b: 9 7 6 18 3 4 8 5 2 1  
c: 2 8 14 15 7 7 20 11 15 19
```

42. Almacene un arreglo de longitud n ordenado en forma ascendente y un elemento x del mismo tipo que los elementos del arreglo, intercalar x en el arreglo v de manera que siga ordenado.



```
public static void main(String[] args) {
    int[] n = {2, 8, 9, 7, 4, 5};
    int[] x = {5, 6, 7, 2, 1, 6};
    int[] z = new int[n.length + x.length];
    int index = 0;
    for (int i = 0; i < n.length; i++) {
        z[index++] = n[i];
        z[index++] = x[i];
    }
    for (int i : z) {
        System.out.print(i + " ");
    }
}
```

2 5 8 6 9 7 7 2 4 1 5 6

43. Almacene 2 arreglos a y b de longitudes n y m, el primero ordenado en forma ascendente y el segundo ordenado de manera descendente, crear un nuevo arreglo c de n + m elementos intercalando los elementos de a y b de modo que c quede ordenado en forma ascendente.

```
int[] b = {10, 10, 10, 10, 2, 4, 6, 8};
int[] a = {3, 5, 61, 1, 9, 12, 4, 2, 9, 15};
int[] c = new int[b.length + a.length];
System.out.print("b: ");
for (int i = b.length - 1; i >= 0; i--) {
    System.out.print(b[i] + " ");
}
System.out.println();
int sum = 0;
for (int i = 0; i < b.length; i++) {
    c[sum++] = b[i];
}
for (int i = 0; i < a.length; i++) {
    c[sum++] = a[i];
}

System.out.print("a: ");
for (int i = 0; i < a.length; i++) {
    System.out.print(a[i] + " ");
}
System.out.println();

System.out.print("c: ");
for (int i = c.length - 1; i >= 0; i--) {
    System.out.print(c[i] + " ");
}
}
```



```
b: 3 6 4 2 19 18 16 18  
a: 3 5 61 1 9 12 4 2 9 15  
c: 15 9 2 4 12 9 1 61 5 3 3 6 4 2 19 18 16 18
```

44. Imprimir las siguientes series:

$1 \times 9 + 2 = 11$	1
$12 \times 9 + 3 = 111$	$1 \times 1 = 1$
$123 \times 9 + 4 = 1111$	$11 \times 11 = 121$
$1234 \times 9 + 5 = 11111$	$111 \times 111 = 12321$
$12345 \times 9 + 6 = 111111$	$1111 \times 1111 = 1234321$
$123456 \times 9 + 7 = 1111111$	$11111 \times 11111 = 123454321$
$1234567 \times 9 + 8 = 11111111$	$111111 \times 111111 = 12345654321$
$12345678 \times 9 + 9 = 111111111$	$1111111 \times 1111111 = 1234567654321$
$123456789 \times 9 + 10 = 1111111111$	$11111111 \times 11111111 = 12345678987654321$

```
public static void main(String[] args) {  
    int num = 1;  
    int edit = 2;  
  
    for (int i = 1; i < 9; i++) {  
        System.out.println(num + " * 9 + " + edit + " = " + (num * 9 + edit));  
        num = num * 10 + (i + 1);  
        edit++;  
    }  
  
    System.out.println();  
  
    int n = 1;  
    for (int i = 1; i < 10; i++) {  
        System.out.println(n + " * " + n + " = " + (n * n));  
        n = n * 10 + 1;  
    }  
}
```

```
run:  
1 * 9 + 2 = 11  
12 * 9 + 3 = 111  
123 * 9 + 4 = 1111  
1234 * 9 + 5 = 11111  
12345 * 9 + 6 = 111111  
123456 * 9 + 7 = 1111111  
1234567 * 9 + 8 = 11111111  
12345678 * 9 + 9 = 111111111  
  
1 * 1 = 1  
11 * 11 = 121
```

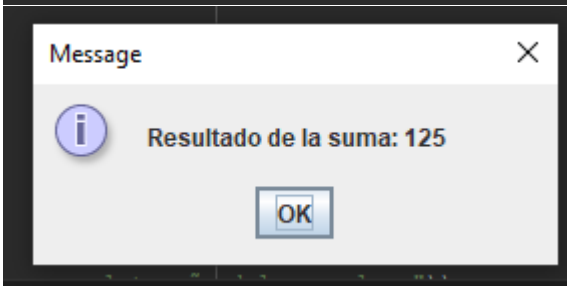
45. Calcular la suma de todos los elementos de un arreglo.

```
public static void main(String[] args) {
    int tam = Integer.parseInt(JOptionPane.showInputDialog("Ingrese el tamaño del arreglo: "));

    int [] numeros = new int [tam];

    int suma = 0;

    for (int i = 0; i < numeros.length; i++){
        numeros[i] = Integer.parseInt(JOptionPane.showInputDialog("Ingrese el número "+i));
        suma = suma + numeros[i];
    }
    JOptionPane.showMessageDialog(null, "Resultado de la suma: "+suma);
}
}
```



46. Realizar un arreglo que registre n palabras y muestre aquellas que inician con una vocal.

```
public static void main(String[] args) {
    String[] palabras = {"arreglo", "programacion", "java", "programador", "computadora"};
    char[] vocales = {'a', 'e', 'i', 'o', 'u'};
    String[] palabrasConVocalInicial = new String[palabras.length];
    int count = 0;
    for (int i = 0; i < palabras.length; i++) {
        char primeraLetra = Character.toLowerCase(palabras[i].charAt(0));
        for (int j = 0; j < vocales.length; j++) {
            if (primeraLetra == vocales[j]) {
                palabrasConVocalInicial[count] = palabras[i];
                count++;
                break;
            }
        }
    }
    System.out.print("Palabras que inician con una vocal: ");
    for (int i = 0; i < count; i++) {
        System.out.print(palabrasConVocalInicial[i] + " ");
    }
}
}
```

run:

Palabras que inician con una vocal: arreglo



47. Una aplicación de las computadoras es dibujar gráficos de barra (llamados histogramas). Escriba un programa que dibuje un gráfico como el siguiente donde la cantidad de asteriscos de cada línea corresponda al número que está a la par.

```
3. | ***
   |
5. | *****
   |
7. | *****
   |
9. | *****
   |_____
```

```
public static void main(String[] args) {
    int[] fila = {2, 15, 7, 8};

    for (int i = 0; i < fila.length; i++) {
        System.out.print(fila[i] + " ");

        for (int j = 0; j < fila[i]; j++) {
            System.out.print("*");
        }

        System.out.println();
    }
}
```

```
run:
2 **
15 *****
7 *****
8 *****
```

48. Escribir un programa que visualice la siguiente salida:

```
1
1 2
1 2 3
1 2 3 4
1 2 3
1 2
1
```



```
public static void main(String[] args) {  
    int[] filas = {1, 2, 3, 4, 3, 2, 1};  
  
    for (int cantidad : filas) {  
        ArrayList<Integer> linea = new ArrayList<>();  
  
        for (int i = 1; i <= cantidad; i++) {  
            linea.add(i);  
        }  
  
        for (int i = 0; i < linea.size(); i++) {  
            System.out.print(linea.get(i) + " ");  
        }  
  
        System.out.println();  
    }  
}
```

```
run:  
1  
1 2  
1 2 3  
1 2 3 4  
1 2 3  
1 2  
1
```

49. Escriba un programa que imprima la forma en diamante siguiente.

```
      *  
     * *  
    * * * *  
   * * * * *  
  * * * * * *  
 * * * * * * *  
* * * * * * * *  
 * * * * * *  
  * * * * *  
   * * * *  
    * * *  
     * *
```

```
public static void main(String[] args) {
    int x = 5;
    int a = 0;

    while (a < x) {
        int esp = x - a;
        while (esp > 0) {
            System.out.print(" ");
            esp--;
        }

        int b = 1;

        while (b <= (2 * a - 1)) {
            System.out.print("* ");
            b++;
        }
        System.out.println();
        a++;
    }
}
```

```
a = x;
while (a > 0) {
    int esp = x - a;
    while (esp > 0) {
        System.out.print(" ");
        esp--;
    }

    int b = 1;
    while (b <= (2 * a - 1)) {
        System.out.print("* ");
        b++;
    }
}
```

```

      *
    * * *
  * * * * *
* * * * * * *
* * * * * * *
  * * * * *
    * * *
      *
```

50. Escriba un programa que imprima los siguientes patrones por separado, uno debajo del siguiente. Utilice ciclos "for" para generar los patrones.



(A)

```
*
**
***
****
*****
*****
*****
*****
*****
*****
*****
*****
*****
```

(B)

```
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
```

(C)

```
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
```

(D)

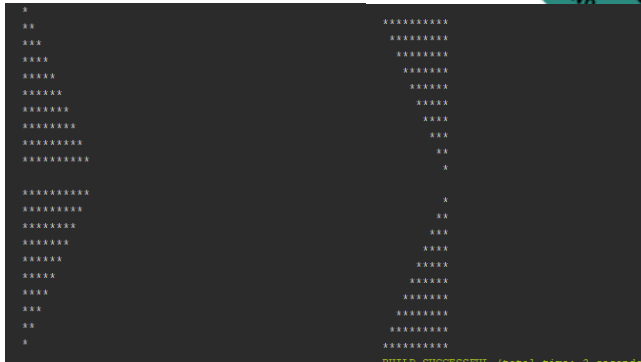
```
*
**
***
****
*****
*****
*****
*****
*****
*****
*****
*****
*****
```

```
char[] arr = new char[n];

for (int i = 1; i <= n; i++) {
    for (int j = 0; j < i; j++) {
        arr[j] = '*';
    }
    for (int j = 0; j < i; j++) {
        System.out.print(arr[j]);
    }
    System.out.println();
}
System.out.println();

for (int i = n; i >= 1; i--) {
    for (int j = 0; j < i; j++) {
        arr[j] = '*';
    }
    for (int j = 0; j < i; j++) {
        System.out.print(arr[j]);
    }
    System.out.println();
}
System.out.println();

for (int i = n; i >= 1; i--) {
    for (int j = 0; j < n - i; j++) {
        System.out.print(" ");
    }
    for (int j = 0; j < i; j++) {
        arr[j] = '*';
    }
    for (int j = 0; j < i; j++) {
        System.out.print(arr[j]);
    }
}
```



51. Un programa que almacene el nombre de 15 alumnos y su número favorito, al final que imprima todos los alumnos con su número favorito, y que aparte imprima los número de menor a mayor.



```
package ejercicio51;
import javax.swing.JOptionPane;
import java.util.Arrays;
public class Ejercicio51 {
    public static void main(String[] args) {
        int x=15;
        String[] alumnos=new String[x];
        int[] numeros=new int[x];

        for (int i=0; i<x; i++) {
            alumnos[i]=JOptionPane.showInputDialog("Ingrese el nombre del alumno "+i+":");
            numeros[i]=Integer.parseInt(JOptionPane.showInputDialog("Ingrese el numero favorito del alumno "+alumnos[i]+":"));
        }

        String mostrar="Lista de alumnos y sus números favoritos:\n";
        for (int i=0; i<x; i++) {
            mostrar=mostrar+alumnos[i]+": "+numeros[i]+\n";
        }
        JOptionPane.showMessageDialog(null, mostrar);

        Arrays.sort(numeros);
        String numsOrdenados="Lista de numeros favoritos ordenados de menor a mayor: \n";
        for (int i=0; i<x; i++) {
            numsOrdenados=numsOrdenados+numeros[i]+", ";
        }
        JOptionPane.showMessageDialog(null, numsOrdenados);
    }
}
```

Lista de numeros favoritos ordenados de menor a mayor:
1, 1, 4, 5, 6, 9, 19, 20, 22, 25, 33, 66, 74, 84, 90,

Lista de alumnos y sus números favoritos:

ana: 1
pepe: 90
lili: 6
nana: 4
sili: 22
ene: 84
leo: 25
saul: 9
jorge: 20
lulu: 5
azul: 1
lala: 66
stiv: 74
lila: 33
ana: 19



52. Una suma de 2 arreglos, validar que no tenga espacios vacíos.

```
package ejercicio52;
import javax.swing.JOptionPane;
public class Ejercicio52 {
    public static void main(String[] args) {
        int x=2;
        int[] arreglo1=new int[x];
        int[] arreglo2=new int[x];
        for (int i=0; i<x; i++){
            String ingresar1=JOptionPane.showInputDialog("Ingrese el numero para la posición "+i+" del arreglo 1");
            if (ingresar1==null || ingresar1.isEmpty()){
                JOptionPane.showMessageDialog(null, "No se permiten espacios vacios");
            }else{
                arreglo1[i]=Integer.parseInt(ingresar1);
            }
        }
        for (int i=0; i<x; i++) {
            String ingresar2=JOptionPane.showInputDialog("Ingrese el numero para la posición "+i+" del arreglo 2");
            if (ingresar2==null || ingresar2.isEmpty()){
                JOptionPane.showMessageDialog(null, "No se permiten espacios vacios");
            }else{
                arreglo2[i]=Integer.parseInt(ingresar2);
            }
        }
        int suma=0;
        for (int i=0; i<x; i++){
            suma=suma+arreglo1[i]+arreglo2[i];
        }
        JOptionPane.showMessageDialog(null, "La suma total de los valores de ambos arreglos es: "+suma);
    }
}
```

La suma total de los valores de ambos arreglos es: 23

53. Realizar un arreglo a partir de un número "x" donde se den los múltiplos de "n" número hasta "y" número.

```
package ejercicio53;
import javax.swing.JOptionPane;
public class Ejercicio53 {
    public static void main(String[] args) {
        int x=Integer.parseInt(JOptionPane.showInputDialog("Ingresa el valor inicial:"));
        int y=Integer.parseInt(JOptionPane.showInputDialog("Ingresa el valor final:"));
        int n=Integer.parseInt(JOptionPane.showInputDialog("Ingresa el numero del que quiere saber sus multiplos:"));
        String resultado="";

        for (int i=x; i<=y; i++){
            if (i%n==0) {
                resultado=resultado+i+" ";
            }
        }

        if (resultado.equals("")){
            JOptionPane.showMessageDialog(null, "No se encontraron multiplos de "+n+" entre "+x+" y "+y);
        }else{
            JOptionPane.showMessageDialog(null, "Los multiplos de "+n+" entre "+x+" y "+y+" son: "+resultado);
        }
    }
}
```

Los multiplos de 5 entre 12 y 33 son: 15 20 25 30



54. Pide al usuario que ingrese 5 nombres y almacénalos en un arreglo, luego muéstralos uno por uno.

Los nombres ingresados son:

- 0.- ari
- 1.- leo
- 2.- rivers
- 3.- cili
- 4.- luzu

```
package ejercicio54;
import javax.swing.JOptionPane;
public class Ejercicio54 {
    public static void main(String[] args) {
        String[] nombres=new String[5];

        for (int i=0; i<nombres.length; i++) {
            nombres[i]=JOptionPane.showInputDialog("Ingrese el nombre de la persona "+i+":");
        }

        String resultado = "Los nombres ingresados son:\n";
        for (int i=0; i<5; i++) {
            resultado=resultado+i+".- "+nombres[i]+" \n";
        }
        JOptionPane.showMessageDialog(null, resultado);
    }
}
```

55. A partir de un arreglo de n numero de valores, contar cuantos números del arreglo son mayores al valor en la posición.

Hay 11 numeros mayores que el valor seleccionado

```
package ejercicio55;
import javax.swing.JOptionPane;
public class Ejercicio55 {
    public static void main(String[] args) {
        int x=Integer.parseInt(JOptionPane.showInputDialog("Ingrese el tamaño del arreglo:"));
        int[] numeros=new int[x];

        for (int i=0; i<x; i++) {
            numeros[i]=Integer.parseInt(JOptionPane.showInputDialog("Ingrese el valor de la posición "+i+":"));
        }

        int posicion=Integer.parseInt(JOptionPane.showInputDialog("Ingrese la posicion del valor que "
            + "quiere comparar: "));
        if (posicion<0 || posicion>=x) {
            JOptionPane.showMessageDialog(null, "Invalido");
        }

        int valorPosicion=numeros[posicion];

        int contador=1;
        for (int i=0; i<x; i++) {
            if (numeros[i]>valorPosicion){
                contador++;
            }
        }
        JOptionPane.showMessageDialog(null, "Hay "+ contador+" numeros mayores que el valor seleccionado");
    }
}
```



56. Realizar un arreglo que pida el nombre de n personas y ordenarlos alfabéticamente..

Lista de personas ordenadas por su nombre en orden alfabético:

ale
azul
blanca
leo
lila
lilas
lili
margarita
mari
rosa

```
package ejercicio56;
import javax.swing.JOptionPane;
import java.util.Arrays;
public class Ejercicio56 {
    public static void main(String[] args) {
        int x=Integer.parseInt(JOptionPane.showInputDialog("Ingrese la cantidad de personas:"));
        String[] personas=new String[x];

        for (int i=0; i<x; i++) {
            personas[i]=JOptionPane.showInputDialog("Ingrese el nombre de la persona "+i+":");
        }

        Arrays.sort(personas);

        String personasAlf="Lista de personas ordenadas por su nombre en orden alfabético: \n";
        for (int i=0; i<x; i++) {
            personasAlf=personasAlf+personas[i]+" \n";
        }
        JOptionPane.showMessageDialog(null, personasAlf);
    }
}
```

57. Escribe un programa que reciba un arreglo de números enteros y determine cuál es el valor máximo del arreglo.

El valor máximo en el arreglo es: 11

```
package ejercicio57;
import javax.swing.JOptionPane;
public class Ejercicio57 {
    public static void main(String[] args) {
        int x=Integer.parseInt(JOptionPane.showInputDialog("Ingrese el tamaño del arreglo: "));

        int[] numeros=new int[x];

        for (int i=0; i<x; i++){
            numeros[i]=Integer.parseInt(JOptionPane.showInputDialog("Ingrese el valor de la posición "+i+":"));
        }
        int max=numeros[0];
        for (int i=1; i<x; i++){
            if (numeros[i]>max){
                max=numeros[i];
            }
        }
        JOptionPane.showMessageDialog(null, "El valor máximo en el arreglo es: "+max);
    }
}
```



58. realizar un programa que te pida una rima, el programa imprimirá el largo de la rima, en caso de que las palabras rimen imprimirlas, en caso de que no rimen salir un mensaje de alerta.

```
package ejercicio59;
import javax.swing.JOptionPane;
public class Ejercicio59 {
    public static void main(String[] args) {
        int x=Integer.parseInt(JOptionPane.showInputDialog("Ingrese la cantidad de personas: "));

        String[] nombres=new String[x];
        double[] estaturas=new double[x];

        for (int i=0; i<x; i++) {
            nombres[i]=JOptionPane.showInputDialog("Ingrese el nombre de la persona "+i+":");
            estaturas[i]=Double.parseDouble(JOptionPane.showInputDialog("Ingrese la estatura de "+nombres[i]
                + " en centimetros:"));
        }
        String listaNombres="Personas registradas:\n";
        for (int i=0; i<x; i++) {
            listaNombres=listaNombres+nombres[i]+" \n";
        }
        JOptionPane.showMessageDialog(null, listaNombres);

        String buscar=JOptionPane.showInputDialog("Ingrese el nombre de la persona para consultar su estatura");
        int i;
        for (i=0; i<x; i++) {
            if (nombres[i].equals(buscar)) {
                JOptionPane.showMessageDialog(null, "La estatura de "+nombres[i]+" es de "+estaturas[i]);
            }else{
                JOptionPane.showMessageDialog(null, "Persona no encontrada");
            }
        }
    }
}
```

Las palabras no riman.

Largo de la primera palabra: 5

Largo de la segunda palabra: 6



59. Un problema el cual almacene el nombre y estatura de una personas, al final que imprima solo el nombre de las personas y que el usuario pueda solicitar la estatura de cualquier persona por medio de su nombre.

Personas registradas:

ana
azul
blanca
rosa
lila
rosa
margarita
margaret
ceci

La estatura de rosa es de 1.5

```
package ejercicio59;
import javax.swing.JOptionPane;
public class Ejercicio59 {
    public static void main(String[] args) {
        int x=Integer.parseInt(JOptionPane.showInputDialog("Ingrese la cantidad de personas: "));

        String[] nombres=new String[x];
        double[] estaturas=new double[x];

        for (int i=0; i<x; i++) {
            nombres[i]=JOptionPane.showInputDialog("Ingrese el nombre de la persona "+i+":");
            estaturas[i]=Double.parseDouble(JOptionPane.showInputDialog("Ingrese la estatura de "+nombres[i]+" "
                + "en centimetros:"));
        }
        String listaNombres="Personas registradas:\n";
        for (int i=0; i<x; i++) {
            listaNombres=listaNombres+nombres[i]+"\\n";
        }
        JOptionPane.showMessageDialog(null, listaNombres);

        String buscar=JOptionPane.showInputDialog("Ingrese el nombre de la persona para consultar su estatura:");
        int i;
        for (i=0; i<x; i++) {
            if (nombres[i].equals(buscar)) {
                JOptionPane.showMessageDialog(null, "La estatura de "+nombres[i]+" es de "+estaturas[i]);
            }else{
                JOptionPane.showMessageDialog(null, "Persona no encontrada");
            }
        }
    }
}
```



60. Realizar un programa que permita ingresar los géneros de videojuegos y que los ordene, solicitando ingresar un número determinado de genero de videojuegos y ordenarlos alfabéticamente.

Lista de generos ordenados por su nombre en orden alfabetico:

call of duty
mortal combat
valorant

```
package ejercicio60;
import javax.swing.JOptionPane;
import java.util.Arrays;
public class Ejercicio60 {
    public static void main(String[] args) {
        int x=Integer.parseInt(JOptionPane.showInputDialog("Ingrese la cantidad de generos de videojuegos:"));

        String[] generos=new String[x];

        for (int i=0; i<x; i++) {
            generos[i]=JOptionPane.showInputDialog("Ingrese el género de videojuego "+i+":");
        }

        Arrays.sort(generos);

        String generosAlf="Lista de generos ordenados por su nombre en orden alfabetico: \n";
        for (int i=0; i<x; i++) {
            generosAlf=generosAlf+generos[i]+\n";
        }
        JOptionPane.showMessageDialog(null, generosAlf);
    }
}
```



61. Realizar un arreglo de 10 posiciones que recopile números de 1 a 10 en arábigo y haga la conversión a números romanos. No es necesario un orden en el registro de valores del arreglo.

Conversion:
Arábigo -> Romano

1 -> I
2 -> II
3 -> III
4 -> IV
5 -> V
6 -> VI
7 -> VII
8 -> VIII
9 -> IX
10 -> X

```
package ejercicio62;
import javax.swing.JOptionPane;
public class Ejercicio62 {
    public static void main(String[] args) {
        String [] meses = {"Enero", "Febrero", "Marzo", "Abril", "Mayo", "Junio", "Julio", "Agosto", "Septiembre", "Octubre", "Noviembre", "Diciembre"};
        double [] ahorros = new double [12];

        double euro = 0.043;
        double dolarA = 0.05;
        double dolarC = 0.068;
        double yenes = 6.88;
        double soles = 0.203;

        double sumArr = 0;
        for(int i = 0; i < meses.length; i++){
            ahorros[i] = Integer.parseInt(JOptionPane.showInputDialog("Ingrese el ahorro para "+meses[i]));
            sumArr = sumArr + ahorros[i];
        }
        System.out.println("Ahorros en Pesos");
        for(int i = 0; i < meses.length; i++){
            System.out.println("Ahorros en el mes de "+meses[i]+": "+ahorros[i]);
        }
        System.out.println("Ahorro Anual: "+sumArr);

        System.out.println("\nAhorros en Euros");
        for(int i = 0; i < meses.length; i++){
            System.out.println("Ahorros en el mes de "+meses[i]+": "+ahorros[i]*euro);
        }
        System.out.println("Ahorro Anual: "+sumArr*euro);
    }
}
```



62. Realizar los arreglos necesarios para almacenar la cantidad ahorrada cada mes del año y al cierre del último mes se muestre el total de ahorro mensual y anual en pesos, euros, dólares americanos, dólares canadienses, yenes y soles.

```
System.out.println("\nAhorros en Dolares Americanos");
for(int i = 0; i < meses.length; i++){
    System.out.println("Ahorros en el mes de "+meses[i]+": "+ahorros[i]*dolarA);
}
System.out.println("Ahorro Anual: "+sumArr*dolarA);

System.out.println("\nAhorros en Dolares Canadienses");
for(int i = 0; i < meses.length; i++){
    System.out.println("Ahorros en el mes de "+meses[i]+": "+ahorros[i]*dolarC);
}
System.out.println("Ahorro Anual: "+sumArr*dolarC);

System.out.println("\nAhorros en Yenes");
for(int i = 0; i < meses.length; i++){
    System.out.println("Ahorros en el mes de "+meses[i]+": "+ahorros[i]*yenes);
}
System.out.println("Ahorro Anual: "+sumArr*yenes);

System.out.println("\nAhorros en Soles");
for(int i = 0; i < meses.length; i++){
    System.out.println("Ahorros en el mes de "+meses[i]+": "+ahorros[i]*soles);
}
System.out.println("Ahorro Anual: "+sumArr*soles);
}
}
```

```
System.out.println("\nAhorros en Dolares Americanos");
for(int i = 0; i < meses.length; i++){
    System.out.println("Ahorros en el mes de "+meses[i]+": "+ahorros[i]*dolarA);
}
System.out.println("Ahorro Anual: "+sumArr*dolarA);

System.out.println("\nAhorros en Dolares Canadienses");
for(int i = 0; i < meses.length; i++){
    System.out.println("Ahorros en el mes de "+meses[i]+": "+ahorros[i]*dolarC);
}
System.out.println("Ahorro Anual: "+sumArr*dolarC);

System.out.println("\nAhorros en Yenes");
for(int i = 0; i < meses.length; i++){
    System.out.println("Ahorros en el mes de "+meses[i]+": "+ahorros[i]*yenes);
}
System.out.println("Ahorro Anual: "+sumArr*yenes);

System.out.println("\nAhorros en Soles");
for(int i = 0; i < meses.length; i++){
    System.out.println("Ahorros en el mes de "+meses[i]+": "+ahorros[i]*soles);
}
System.out.println("Ahorro Anual: "+sumArr*soles);
}
}
```



Ahorros en Soles

Ahorros en el mes de Enero: 182.70000000000002
Ahorros en el mes de Febrero: 121.80000000000001
Ahorros en el mes de Marzo: 142.10000000000002
Ahorros en el mes de Abril: 203.0
Ahorros en el mes de Mayo: 101.5
Ahorros en el mes de Junio: 60.900000000000006
Ahorros en el mes de Julio: 20.3
Ahorros en el mes de Agosto: 162.4
Ahorros en el mes de Septiembre: 192.85000000000002
Ahorros en el mes de Octubre: 127.89000000000001
Ahorros en el mes de Noviembre: 60.900000000000006
Ahorros en el mes de Diciembre: 81.2
Ahorro Anual: 1457.5400000000002

Ahorros en Dolares Canadienses

Ahorros en el mes de Enero: 61.2
Ahorros en el mes de Febrero: 40.800000000000004
Ahorros en el mes de Marzo: 47.6
Ahorros en el mes de Abril: 68.0
Ahorros en el mes de Mayo: 34.0
Ahorros en el mes de Junio: 20.400000000000002
Ahorros en el mes de Julio: 6.800000000000001
Ahorros en el mes de Agosto: 54.400000000000006
Ahorros en el mes de Septiembre: 64.60000000000001
Ahorros en el mes de Octubre: 42.84
Ahorros en el mes de Noviembre: 20.400000000000002
Ahorros en el mes de Diciembre: 27.200000000000003
Ahorro Anual: 488.24

Ahorros en Pesos

Ahorros en el mes de Enero: 900.0
Ahorros en el mes de Febrero: 600.0
Ahorros en el mes de Marzo: 700.0
Ahorros en el mes de Abril: 1000.0
Ahorros en el mes de Mayo: 500.0
Ahorros en el mes de Junio: 300.0
Ahorros en el mes de Julio: 100.0
Ahorros en el mes de Agosto: 800.0
Ahorros en el mes de Septiembre: 950.0
Ahorros en el mes de Octubre: 630.0
Ahorros en el mes de Noviembre: 300.0
Ahorros en el mes de Diciembre: 400.0
Ahorro Anual: 7180.0

Ahorros en el mes de Marzo: 35.0
Ahorros en el mes de Abril: 50.0
Ahorros en el mes de Mayo: 25.0
Ahorros en el mes de Junio: 15.0
Ahorros en el mes de Julio: 5.0
Ahorros en el mes de Agosto: 40.0
Ahorros en el mes de Septiembre: 47.5
Ahorros en el mes de Octubre: 31.5
Ahorros en el mes de Noviembre: 15.0
Ahorros en el mes de Diciembre: 20.0
Ahorro Anual: 359.0

Ahorros en Yenes

Ahorros en el mes de Enero: 6192.0
Ahorros en el mes de Febrero: 4128.0
Ahorros en el mes de Marzo: 4816.0
Ahorros en el mes de Abril: 6880.0
Ahorros en el mes de Mayo: 3440.0
Ahorros en el mes de Junio: 2064.0
Ahorros en el mes de Julio: 688.0
Ahorros en el mes de Agosto: 5504.0
Ahorros en el mes de Septiembre: 6536.0
Ahorros en el mes de Octubre: 4334.4
Ahorros en el mes de Noviembre: 2064.0
Ahorros en el mes de Diciembre: 2752.0
Ahorro Anual: 49398.4

Ahorros en Euros

Ahorros en el mes de Enero: 38.699999999999996
Ahorros en el mes de Febrero: 25.799999999999997
Ahorros en el mes de Marzo: 30.099999999999998
Ahorros en el mes de Abril: 43.0
Ahorros en el mes de Mayo: 21.5
Ahorros en el mes de Junio: 12.899999999999999
Ahorros en el mes de Julio: 4.3
Ahorros en el mes de Agosto: 34.4
Ahorros en el mes de Septiembre: 40.849999999999994
Ahorros en el mes de Octubre: 27.089999999999996
Ahorros en el mes de Noviembre: 12.899999999999999
Ahorros en el mes de Diciembre: 17.2
Ahorro Anual: 308.73999999999995

Ahorros en Euros

Ahorros en el mes de Enero: 38.699999999999996
Ahorros en el mes de Febrero: 25.799999999999997
Ahorros en el mes de Marzo: 30.099999999999998
Ahorros en el mes de Abril: 43.0
Ahorros en el mes de Mayo: 21.5
Ahorros en el mes de Junio: 12.899999999999999
Ahorros en el mes de Julio: 4.3
Ahorros en el mes de Agosto: 34.4
Ahorros en el mes de Septiembre: 40.849999999999994
Ahorros en el mes de Octubre: 27.089999999999996
Ahorros en el mes de Noviembre: 12.899999999999999
Ahorros en el mes de Diciembre: 17.2
Ahorro Anual: 308.73999999999995



63. Realizar un programa que almacene una palabra, la descomponga en un arreglo y muestre la cantidad de vocales y consonantes que la conforman.

```
package ejercicio63;
import javax.swing.*;
import java.util.Arrays;
public class Ejercicio63 {
    public static void main(String[] args) {
        String palabra = JOptionPane.showInputDialog("Ingrese una palabra: ").toLowerCase();
        int v = 0;
        int c = 0;
        char[] letras = palabra.toCharArray();

        for (char letra : letras) {
            if (letra == 'a' || letra == 'e' || letra == 'i' || letra == 'o' || letra == 'u') {
                v++;
            } else if (Character.isLetter(letra)) {
                c++;
            }
        }

        String arreLe = Arrays.toString(letras);
        System.out.println("Palabra descompuesta en un arreglo: " + arreLe);
        System.out.println("Cantidad de vocales: " + v);
        System.out.println("Cantidad de consonantes: " + c);
    }
}
```

```
Palabra descompuesta en un arreglo: [b, r, a, z, o]
Cantidad de vocales: 2
Cantidad de consonantes: 3
```