

Estructura de repetición for

LOGRO DE LA UNIDAD DE APRENDIZAJE

Al finalizar la unidad, el alumno, mediante el uso de estructuras de repetición, implementa programas en Java que resuelven problemas concretos.

TEMARIO

1. Estructura de repetición for

ACTIVIDADES

Los alumnos desarrollan algoritmos que involucren el uso de estructuras de repetición.

1. Estructura de repetición for

La estructura de repetición while es una estructura de propósito general: puede usarse para resolver cualquier problema que involucre procesos repetitivos. Pero, para los casos en que se conoce el número de iteraciones de un proceso repetitivo, la estructura ideal es la estructura for. Fue construida para ese tipo de procesos y es por ello que puede manejar uno o más contadores propios. En la Figura 1, se muestra el diagrama de flujo de la estructura for.

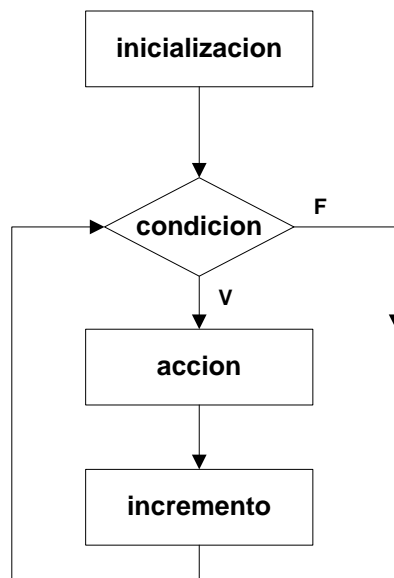


Figura 1 Diagrama de flujo de la estructura **for**

La sintaxis de la estructura de repetición **for** es la siguiente:

Para una sólo acción por repetir:

```
for (inicio; condicion; incremento)
    accion;
```

Para más de una acción por repetir:

```
for (inicio; condicion; incremento) {
    accion1;
    accion2;
    .
    .
    .
    accionn;
}
```

Funcionamiento del for

- **Paso 1:-** Se ejecuta la sentencia de **inicio**. En la sentencia de **inicio**, se declara y se inicializa el contador del **for**.

- **Paso 2:-** Se evalúa la **condicion**. En caso de que la **condicion** sea verdadera, va al paso 3; en caso contrario, finaliza el **for**.
- **Paso 3:-** Se ejecuta la **accion** o el conjunto de **acciones**.
- **Paso 4:-** Se ejecuta la sentencia de **incremento** y vuelve al paso 2. En la sentencia de **incremento** se incrementa el contador del **for**.

Ejemplo 1

- **for** que imprime los números: 0, 1, 2, 3, ..., 48, 49, 50

```
for (int i = 0; i <= 50; i++)
    txtS.append(i + "\n");
```

- **for** que imprime los números: 100, 99, 98, ..., 13, 12, 11, 10

```
for (int i = 100; i >= 10; i--)
    txtS.append( i + "\n" );
```

- **for** que imprime los números: 10, 12, 14, 16, ..., 98, 99, 100

```
for (int i = 10; i <= 100; i += 2)
    txtS.append(i + "\n");
```

- **for** que imprime los números: 100, 97, 94, 91, ..., 18, 15, 12, 9

```
for (int i = 100; i >= 9; i -= 3)
    txtS.append(i + "\n");
```

- **for** que imprime la siguiente tabla:

| | |
|----|----|
| 10 | 30 |
| 11 | 29 |
| 12 | 28 |
| 13 | 27 |
| 14 | 26 |
| 15 | 25 |
| 16 | 24 |
| 17 | 23 |
| 18 | 22 |
| 19 | 21 |
| 20 | 20 |

```
for (int i = 10, j = 30; i <= 20; i++, j--)
    txtS.append(i + "\t" + j + "\n");
```

Nota

El contador del **for** puede ser declarado dentro del mismo **for**, en cuyo caso se considera como variable local al **for**, no siendo accesible fuera de él. De esta manera, en un mismo método, dos o más **for**, no anidados, pueden declarar contadores con el mismo nombre. Cada contador existe dentro del **for** en el que fue declarado.

2. Números aleatorios

Para generar un número aleatorio entero en el intervalo de **min** a **max**, se usa la siguiente expresión:

```
n =(Entero)((max-min+1)*Math.random() + min)
```

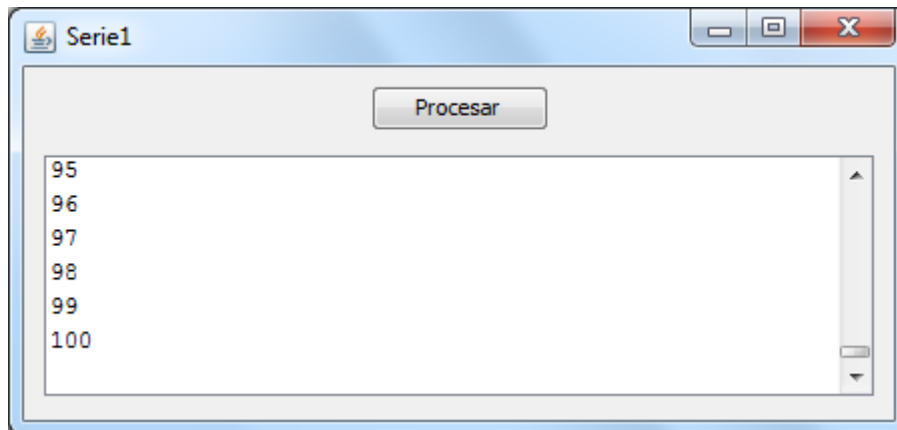
3. Problemas resueltos

Problema 1

Diseñe un programa que imprima y sume la siguiente serie:

1, 2, 3, ..., 100

Solución



```
package cibertec;

import java.awt.EventQueue;

public class Serie1 extends JFrame implements ActionListener {
    private static final long serialVersionUID = 1L;
    private JButton btnProcesar;
    private JScrollPane scpScroll;
    private JTextArea txtS;

    // Lanza la aplicación
    public static void main(String[] args) {
        try {
            UIManager.setLookAndFeel(
                "com.sun.java.swing.plaf.windows.WindowsLookAndFeel");
        }
        catch (Throwable e) {
            e.printStackTrace();
        }
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Serie1 frame = new Serie1();

```

```

        frame.setVisible(true);
    }
    catch (Exception e) {
        e.printStackTrace();
    }
}

});
}

// Crea la GUI
public Serie1() {
    setTitle("Serie1");
    setBounds(100, 100, 450, 214);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    getContentPane().setLayout(null);

    btnProcesar = new JButton("Procesar");
    btnProcesar.addActionListener(this);
    btnProcesar.setBounds(173, 9, 89, 23);
    getContentPane().add(btnProcesar);

    scpScroll = new JScrollPane();
    scpScroll.setBounds(10, 44, 414, 120);
    getContentPane().add(scpScroll);

    txtS = new JTextArea();
    txtS.setFont(new Font("Monospaced", Font.PLAIN, 12));
    scpScroll.setViewportView(txtS);
}

// Direcciona eventos de tipo ActionEvent
public void actionPerformed(ActionEvent arg0) {
    if (arg0.getSource() == btnProcesar) {
        actionPerformedBtnProcesar(arg0);
    }
}

// Procesa la pulsación del botón Procesar
protected void actionPerformedBtnProcesar(ActionEvent arg0) {
    // Declara e inicializa la suma
    int sum = 0;

    // Imprime y suma la serie
    for(int ter = 1; ter <= 100; ter++) {

        // Imprime el término actual
        txtS.append(ter + "\n");

        // Suma el término actual
        sum += ter;
    }

    // Imprime la suma
    txtS.append("Suma : " + sum);
}
}

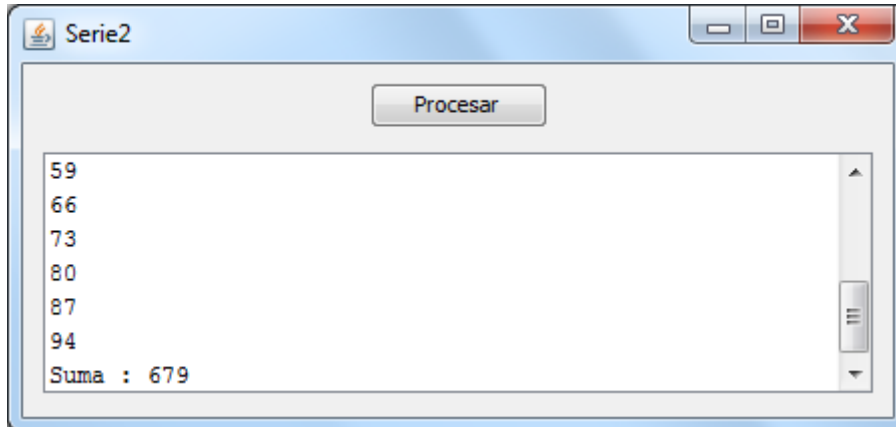
```

Problema 2

Diseñe un programa que imprima y sume la siguiente serie:

3, 10, 17, 24, 31, ..., 94

Solución



```
package cibertec;

import java.awt.EventQueue;
import java.awt.Font;
import javax.swing.JFrame;
import javax.swing.JButton;
import javax.swing.UIManager;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;

public class Serie2 extends JFrame implements ActionListener {
    private static final long serialVersionUID = 1L;
    private JButton btnProcesar;
    private JScrollPane scpScroll;
    private JTextArea txtS;

    // Lanza la aplicación
    public static void main(String[] args) {
        try {
            UIManager.setLookAndFeel(
                "com.sun.java.swing.plaf.windows.WindowsLookAndFeel");
        }
        catch (Throwable e) {
            e.printStackTrace();
        }
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Serie2 frame = new Serie2();
                    frame.setVisible(true);
                }
                catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }
}
```

```

    }
    });
}

// Crea la GUI
public Serie2() {
    setTitle("Serie2");
    setBounds(100, 100, 450, 214);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    getContentPane().setLayout(null);

    btnProcesar = new JButton("Procesar");
    btnProcesar.addActionListener(this);
    btnProcesar.setBounds(173, 9, 89, 23);
    getContentPane().add(btnProcesar);

    scpScroll = new JScrollPane();
    scpScroll.setBounds(10, 44, 414, 120);
    getContentPane().add(scpScroll);

    txtS = new JTextArea();
    txtS.setFont(new Font("Monospaced", Font.PLAIN, 12));
    scpScroll.setViewportView(txtS);
}

// Direcciona eventos de tipo ActionEvent
public void actionPerformed(ActionEvent arg0) {
    if (arg0.getSource() == btnProcesar) {
        actionPerformedBtnProcesar(arg0);
    }
}

// Procesa la pulsación del botón Procesar
protected void actionPerformedBtnProcesar(ActionEvent arg0) {
    // Declara e inicializa la suma
    int sum = 0;

    // Imprime y suma la serie
    for (int ter = 3; ter <= 94; ter += 7) {

        // Imprime el término actual
        txtS.append(ter + "\n");

        // Suma el término actual
        sum += ter;
    }

    // Imprime la suma
    txtS.append("Suma : " + sum);
}
}

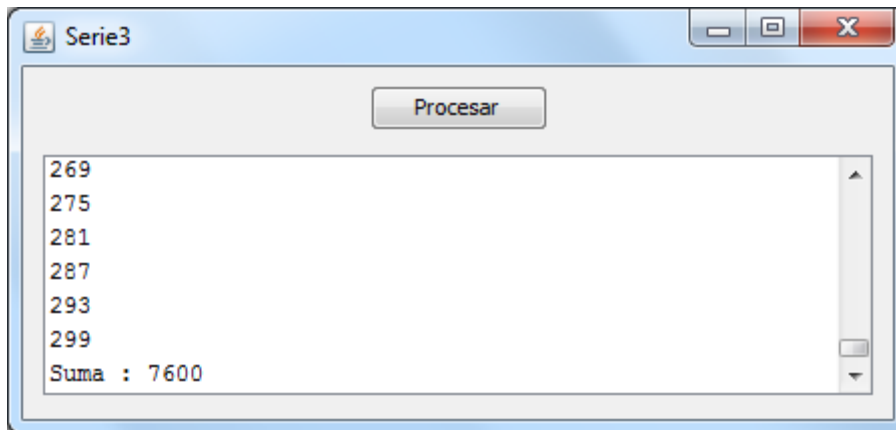
```

Problema 3

Diseña un programa que imprima y sume 50 términos de la siguiente serie:

5, 11, 17, 23, 29, 35, ...

Solución



```
package cibertec;

import java.awt.EventQueue;
import java.awt.Font;
import javax.swing.JFrame;
import javax.swing.JButton;
import javax.swing.UIManager;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;

public class Serie3 extends JFrame implements ActionListener {
    private static final long serialVersionUID = 1L;
    private JButton btnProcesar;
    private JScrollPane scpScroll;
    private JTextArea txtS;

    // Lanza la aplicación
    public static void main(String[] args) {
        try {
            UIManager.setLookAndFeel(
                "com.sun.java.swing.plaf.windows.WindowsLookAndFeel");
        }
        catch (Throwable e) {
            e.printStackTrace();
        }
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Serie3 frame = new Serie3();
                    frame.setVisible(true);
                }
                catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    // Crea la GUI
    public Serie3() {
```



```

setTitle("Serie3");
setBounds(100, 100, 450, 214);
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
getContentPane().setLayout(null);

btnProcesar = new JButton("Procesar");
btnProcesar.addActionListener(this);
btnProcesar.setBounds(173, 9, 89, 23);
getContentPane().add(btnProcesar);

scpScroll = new JScrollPane();
scpScroll.setBounds(10, 44, 414, 120);
getContentPane().add(scpScroll);

txtS = new JTextArea();
txtS.setFont(new Font("Monospaced", Font.PLAIN, 12));
scpScroll.setViewportView(txtS);
}

// Direcciona eventos de tipo ActionEvent
public void actionPerformed(ActionEvent arg0) {
    if (arg0.getSource() == btnProcesar) {
        actionPerformedBtnProcesar(arg0);
    }
}

// Procesa la pulsación del botón Procesar
protected void actionPerformedBtnProcesar(ActionEvent arg0) {
    // Declara e inicializa el término y la suma
    int ter = 5, sum = 0;

    // Imprime y suma la serie
    for (int c = 0; c < 50; c++) {

        // Imprime el término actual
        txtS.append(ter + "\n");

        // Suma el término actual
        sum += ter;

        // Pasa al siguiente término
        ter += 6;
    }

    // Imprime la suma
    txtS.append("Suma : " + sum);
}
}

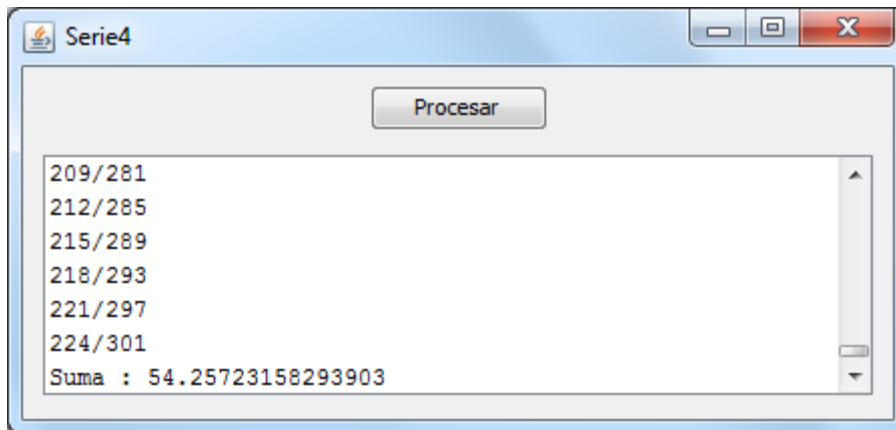
```

Problema 4

Diseñe un programa que imprima y sume 75 términos de la siguiente serie:

$$\frac{2}{5}, \frac{5}{9}, \frac{8}{13}, \frac{11}{17}, \dots$$

Solución



```
package cibertec;

import java.awt.EventQueue;
import java.awt.Font;
import javax.swing.JFrame;
import javax.swing.JButton;
import javax.swing.UIManager;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;

public class Serie4 extends JFrame implements ActionListener {
    private static final long serialVersionUID = 1L;
    private JButton btnProcesar;
    private JScrollPane scpScroll;
    private JTextArea txtS;

    // Lanza la aplicación
    public static void main(String[] args) {
        try {
            UIManager

.setLookAndFeel("com.sun.java.swing.plaf.windows.WindowsLookAndFeel");
        }
        catch (Throwable e) {
            e.printStackTrace();
        }
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Serie4 frame = new Serie4();
                    frame.setVisible(true);
                }
                catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    // Crea la GUI
```

```

public Serie4() {
    setTitle("Serie4");
    setBounds(100, 100, 450, 214);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    getContentPane().setLayout(null);

    btnProcesar = new JButton("Procesar");
    btnProcesar.addActionListener(this);
    btnProcesar.setBounds(173, 9, 89, 23);
    getContentPane().add(btnProcesar);

    scpScroll = new JScrollPane();
    scpScroll.setBounds(10, 44, 414, 120);
    getContentPane().add(scpScroll);

    txtS = new JTextArea();
    txtS.setFont(new Font("Monospaced", Font.PLAIN, 12));
    scpScroll.setViewportView(txtS);
}

// Direcciona eventos de tipo ActionEvent
public void actionPerformed(ActionEvent arg0) {
    if (arg0.getSource() == btnProcesar) {
        actionPerformedBtnProcesar(arg0);
    }
}

// Procesa la pulsación del botón Procesar
protected void actionPerformedBtnProcesar(ActionEvent arg0) {
    // Declara e inicializa el término y la suma
    int num = 2, den = 5;
    double sum = 0;

    // Imprime y suma la serie
    for (int c = 0; c < 75; c++) {

        // Imprime el término actual
        txtS.append(num + "/" + den + "\n");

        // Suma el término actual
        sum += num * 1.0 / den;

        // Pasa al siguiente término
        num += 3;
        den += 4;
    }

    // Imprime la suma
    txtS.append("Suma : " + sum);
}
}

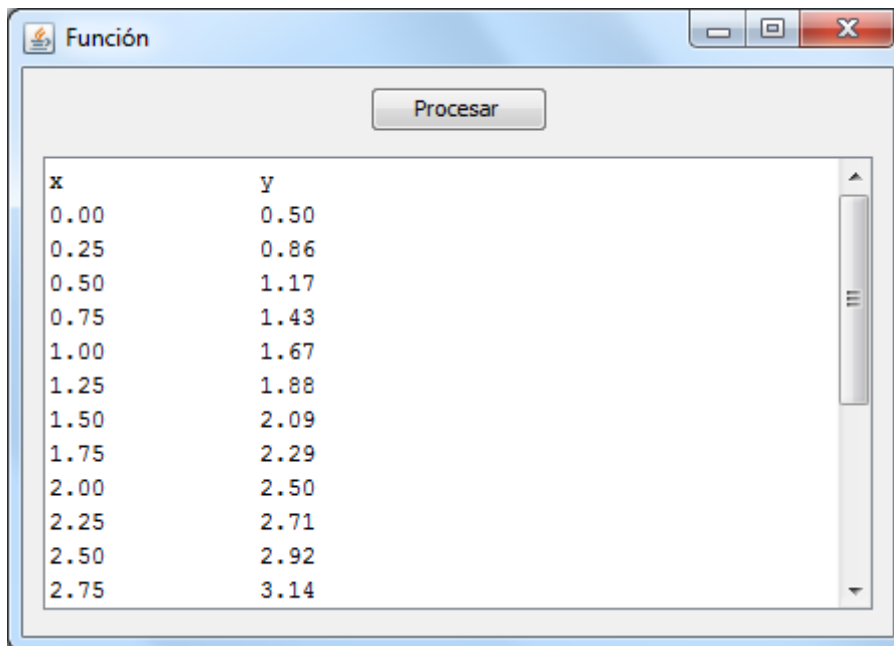
```

Problema 5

Diseña un programa que imprima una tabla de valores de x e y , para valores de x en el intervalo de 0 a 2.75 cada 0.25, siendo:

$$y = \frac{x^3 + 3x + 1}{x^2 + 2}$$

Solución



```
package cibertec;

import java.awt.EventQueue;
import java.awt.Font;
import javax.swing.JFrame;
import javax.swing.JButton;
import javax.swing.UIManager;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import java.util.Locale;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;

public class Funcion extends JFrame implements ActionListener {
    private static final long serialVersionUID = 1L;
    private JButton btnProcesar;
    private JScrollPane scpScroll;
    private JTextArea txtS;

    // Lanza la aplicación
    public static void main(String[] args) {
        try {
            UIManager.setLookAndFeel(
                "com.sun.java.swing.plaf.windows.WindowsLookAndFeel");
        }
        catch (Throwable e) {
            e.printStackTrace();
        }
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Funcion frame = new Funcion();

```

```

        frame.setVisible(true);
    }
    catch (Exception e) {
        e.printStackTrace();
    }
}

});
}

// Crea la GUI
public Funcion() {
    setTitle("Función");
    setBounds(100, 100, 450, 321);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    getContentPane().setLayout(null);

    btnProcesar = new JButton("Procesar");
    btnProcesar.addActionListener(this);
    btnProcesar.setBounds(173, 9, 89, 23);
    getContentPane().add(btnProcesar);

    scpScroll = new JScrollPane();
    scpScroll.setBounds(10, 44, 414, 226);
    getContentPane().add(scpScroll);

    txtS = new JTextArea();
    txtS.setFont(new Font("Monospaced", Font.PLAIN, 12));
    scpScroll.setViewportView(txtS);
}

// Direcciona eventos de tipo ActionEvent
public void actionPerformed(ActionEvent arg0) {
    if (arg0.getSource() == btnProcesar) {
        actionPerformedBtnProcesar(arg0);
    }
}

// Procesa la pulsación del botón Procesar
protected void actionPerformedBtnProcesar(ActionEvent arg0) {
    // Declara e inicializa variables
    double y;

    // Imprime la cabecera
    txtS.append(String.format(Locale.US, "%-15.2s%-15.2s\n", "x", "y"));

    // Imprime la tabla
    for (double x = 0; x <= 2.75; x += 0.25) {

        // Calcula el valor de y para el valor actual de x
        y = (x * x * x + 3 * x + 1) / (x * x + 2);

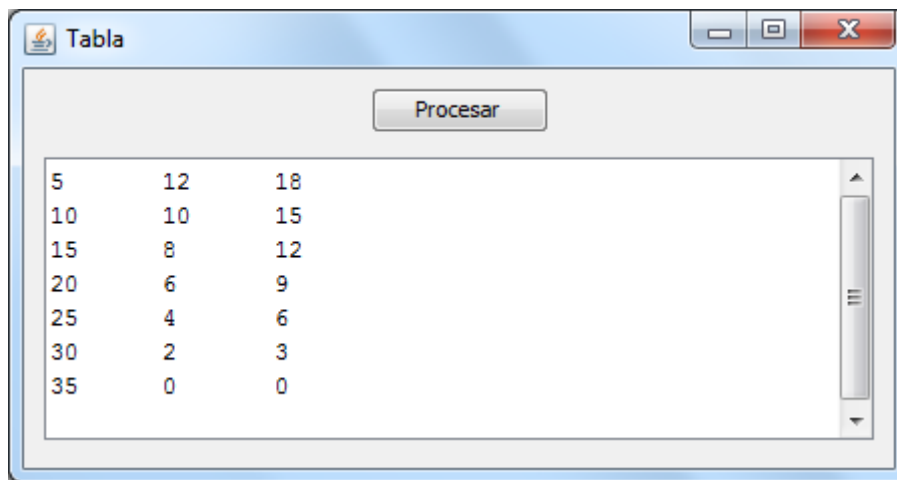
        // Imprime la pareja de valores x e y
        txtS.append(String.format(Locale.US, "%-15.2f%-15.2f\n", x, y));
    }
}
}

```

Problema 6

Diseñe un programa que imprima la siguiente tabla de números:

| | | |
|----|----|----|
| 5 | 12 | 18 |
| 10 | 10 | 15 |
| 15 | 8 | 12 |
| 20 | 6 | 9 |
| 25 | 4 | 6 |
| 30 | 2 | 3 |
| 35 | 0 | 0 |

Solución

```
package cibertec;

import java.awt.EventQueue;
import java.awt.Font;
import javax.swing.JFrame;
import javax.swing.JButton;
import javax.swing.UIManager;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;

public class Tabla extends JFrame implements ActionListener {
    private static final long serialVersionUID = 1L;
    private JButton btnProcesar;
    private JScrollPane scpScroll;
    private JTextArea txtS;

    // Lanza la aplicación
    public static void main(String[] args) {
        try {
            UIManager

.setLookAndFeel("com.sun.java.swing.plaf.windows.WindowsLookAndFeel");
        }
        catch (Throwable e) {
            e.printStackTrace();
        }
    }
}
```

```

    }
    EventQueue.invokeLater(new Runnable() {
        public void run() {
            try {
                Tabla frame = new Tabla();
                frame.setVisible(true);
            }
            catch (Exception e) {
                e.printStackTrace();
            }
        }
    });
}

// Crea la GUI
public Tabla() {
    setTitle("Tabla");
    setBounds(100, 100, 450, 237);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    getContentPane().setLayout(null);

    btnProcesar = new JButton("Procesar");
    btnProcesar.addActionListener(this);
    btnProcesar.setBounds(173, 9, 89, 23);
    getContentPane().add(btnProcesar);

    scpScroll = new JScrollPane();
    scpScroll.setBounds(10, 44, 414, 141);
    getContentPane().add(scpScroll);

    txtS = new JTextArea();
    txtS.setFont(new Font("Monospaced", Font.PLAIN, 12));
    scpScroll.setViewportView(txtS);
}

// Direcciona eventos de tipo ActionEvent
public void actionPerformed(ActionEvent arg0) {
    if (arg0.getSource() == btnProcesar) {
        actionPerformedBtnProcesar(arg0);
    }
}

// Procesa la pulsación del botón Procesar
protected void actionPerformedBtnProcesar(ActionEvent arg0) {
    // Declara e inicializa las columnas
    int colum1 = 5, colum2 = 12, colum3 = 18;

    // Imprime la tabla
    for (int veces = 0; veces < 7; veces++){

        // Imprime una fila de la tabla
        txtS.append(colum1 + "\t" + colum2 + "\t" + colum3 + "\n");

        // Pasa a la siguiente fila
        colum1 += 5;
        colum2 -= 2;
        colum3 -= 3;
    }
}

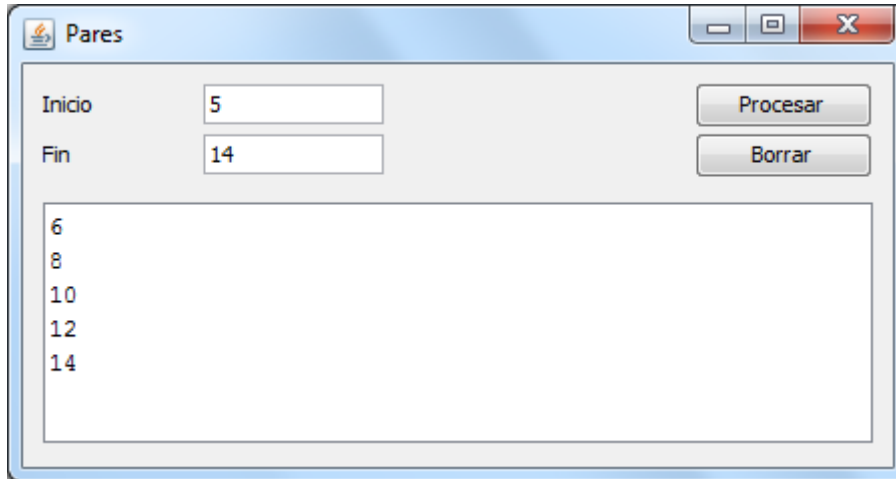
```

```
}
```

Problema 7

Diseñe un programa que lea los extremos de un intervalo de números enteros e imprima todos los números pares del intervalo.

Solución



```
package cibertec;

import java.awt.EventQueue;
import java.awt.Font;
import javax.swing.JFrame;
import javax.swing.JButton;
import javax.swing.UIManager;
import javax.swing.JLabel;
import javax.swing.JTextField;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;

public class Pares extends JFrame implements ActionListener {
    private static final long serialVersionUID = 1L;

    // Declaración de variables
    private JLabel lblInicio;
    private JLabel lblFin;
    private JTextField txtInicio;
    private JTextField txtFin;
    private JButton btnProcesar;
    private JButton btnBorrar;
    private JScrollPane scpScroll;
    private JTextArea txtS;

    // Lanza la aplicación
    public static void main(String[] args) {
        try {
            UIManager.setLookAndFeel(
                "com.sun.java.swing.plaf.windows.WindowsLookAndFeel");
        }
    }
}
```



```

        catch (Throwable e) {
            e.printStackTrace();
        }
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Pares frame = new Pares();
                    frame.setVisible(true);
                }
                catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    // Crea la GUI
    public Pares() {
        setTitle("Pares");
        setBounds(100, 100, 450, 239);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        getContentPane().setLayout(null);

        lblInicio = new JLabel("Inicio");
        lblInicio.setBounds(10, 13, 80, 14);
        getContentPane().add(lblInicio);

        lblFin = new JLabel("Fin");
        lblFin.setBounds(10, 38, 80, 14);
        getContentPane().add(lblFin);

        txtInicio = new JTextField();
        txtInicio.setBounds(90, 10, 90, 20);
        getContentPane().add(txtInicio);
        txtInicio.setColumns(10);

        txtFin = new JTextField();
        txtFin.setBounds(90, 35, 90, 20);
        getContentPane().add(txtFin);
        txtFin.setColumns(10);

        btnProcesar = new JButton("Procesar");
        btnProcesar.addActionListener(this);
        btnProcesar.setBounds(335, 9, 89, 23);
        getContentPane().add(btnProcesar);

        btnBorrar = new JButton("Borrar");
        btnBorrar.addActionListener(this);
        btnBorrar.setBounds(335, 34, 89, 23);
        getContentPane().add(btnBorrar);

        scpScroll = new JScrollPane();
        scpScroll.setBounds(10, 69, 414, 120);
        getContentPane().add(scpScroll);

        txtS = new JTextArea();
        txtS.setFont(new Font("Monospaced", Font.PLAIN, 12));
        scpScroll.setViewportView(txtS);
    }

```

```

// Direcciona eventos de tipo ActionEvent
public void actionPerformed(ActionEvent arg0) {
    if (arg0.getSource() == btnProcesar) {
        actionPerformedBtnProcesar(arg0);
    }
    if (arg0.getSource() == btnBorrar) {
        actionPerformedBtnBorrar(arg0);
    }
}

// Procesa la pulsación del botón Procesar
protected void actionPerformedBtnProcesar(ActionEvent arg0) {
    // Declara variables
    int inicio, fin;

    // Ingresar los extremos del intervalo
    inicio = Integer.parseInt(txtInicio.getText());
    fin = Integer.parseInt(txtFin.getText());

    // Imprime la lista de pares
    for (int numero = inicio; numero <= fin; numero++) {
        if (numero % 2 == 0)
            txtS.append(numero + "\n");
    }
}

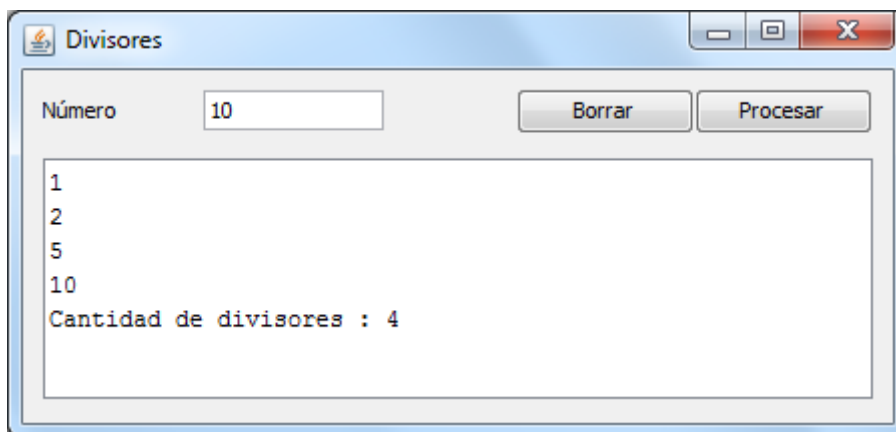
// Procesa la pulsación del botón Borrar
protected void actionPerformedBtnBorrar(ActionEvent arg0) {
    txtInicio.setText("");
    txtFin.setText("");
    txtS.setText("");
    txtInicio.requestFocus();
}
}

```

Problema 8

Diseña un programa que imprima los divisores de un número natural y la cantidad de divisores encontrados.

Solución



```

package cibertec;

import java.awt.EventQueue;
import java.awt.Font;
import javax.swing.JFrame;
import javax.swing.JButton;
import javax.swing.UIManager;
import javax.swing.JLabel;
import javax.swing.JTextField;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;

public class Divisores extends JFrame implements ActionListener {
    private static final long serialVersionUID = 1L;

    // Declaración de variables
    private JLabel lblNumero;
    private JTextField txtNumero;
    private JButton btnProcesar;
    private JButton btnBorrar;
    private JScrollPane scpScroll;
    private JTextArea txtS;

    // Lanza la aplicación
    public static void main(String[] args) {
        try {
            UIManager.setLookAndFeel(
                "com.sun.java.swing.plaf.windows.WindowsLookAndFeel");
        }
        catch (Throwable e) {
            e.printStackTrace();
        }
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Divisores frame = new Divisores();
                    frame.setVisible(true);
                }
                catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    // Crea la GUI
    public Divisores() {
        setTitle("Divisores");
        setBounds(100, 100, 450, 214);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        getContentPane().setLayout(null);

        lblNumero = new JLabel("Número");
        lblNumero.setBounds(10, 13, 80, 14);
        getContentPane().add(lblNumero);

        txtNumero = new JTextField();

```

```

txtNumero.setBounds(90, 10, 90, 20);
getContentPane().add(txtNumero);
txtNumero.setColumns(10);

btnProcesar = new JButton("Procesar");
btnProcesar.addActionListener(this);
btnProcesar.setBounds(335, 9, 89, 23);
getContentPane().add(btnProcesar);

btnBorrar = new JButton("Borrar");
btnBorrar.addActionListener(this);
btnBorrar.setBounds(246, 9, 89, 23);
getContentPane().add(btnBorrar);

scpScroll = new JScrollPane();
scpScroll.setBounds(10, 44, 414, 120);
getContentPane().add(scpScroll);

txtS = new JTextArea();
txtS.setFont(new Font("Monospaced", Font.PLAIN, 12));
scpScroll.setViewportView(txtS);
}

// Direcciona eventos de tipo ActionEvent
public void actionPerformed(ActionEvent arg0) {
    if (arg0.getSource() == btnProcesar) {
        actionPerformedBtnProcesar(arg0);
    }
    if (arg0.getSource() == btnBorrar) {
        actionPerformedBtnBorrar(arg0);
    }
}

// Procesa la pulsación del botón Procesar
protected void actionPerformedBtnProcesar(ActionEvent arg0) {
    // Declara variables
    int contadiv = 0, numero;

    // Ingresar el número
    numero = Integer.parseInt(txtNumero.getText());

    // Imprime los divisores del número
    for (int divisor = 1; divisor <= numero; divisor++) {

        // Si divisor divide a numero de forma exacta, es un divisor
        if (numero % divisor == 0) {

            // Imprime el divisor encontrado
            txtS.append(divisor + "\n");

            // Cuenta el divisor encontrado
            contadiv++;
        }
    }

    // Imprime la cantidad de divisores
    txtS.append("Cantidad de divisores : " + contadiv);
}

```

```

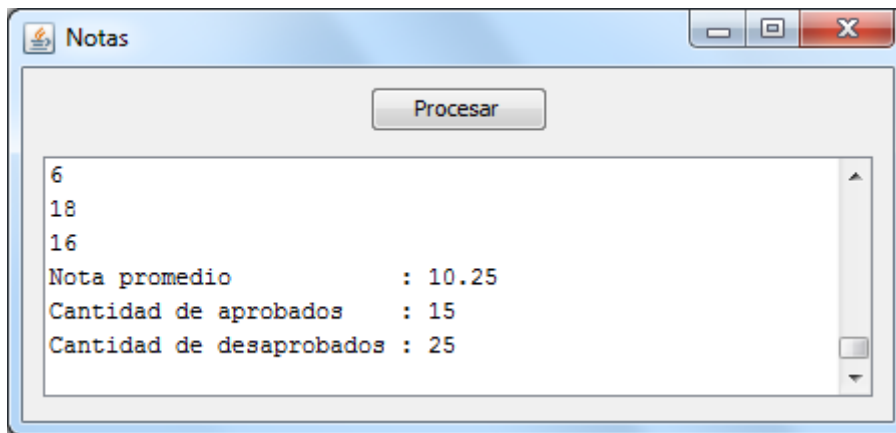
// Procesa la pulsación del botón Borrar
protected void actionPerformedBtnBorrar(ActionEvent arg0) {
    txtNumero.setText("");
    txtS.setText("");
    txtNumero.requestFocus();
}
}

```

Problema 9

Diseñe un programa que genere aleatoriamente las notas de un examen de 40 estudiantes de una sección y determine la nota promedio, la cantidad de notas aprobatorias y la cantidad de notas desaprobatorias

Solución



```

package cibertec;

import java.awt.EventQueue;
import java.awt.Font;
import javax.swing.JFrame;
import javax.swing.JButton;
import javax.swing.UIManager;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;

public class Notas extends JFrame implements ActionListener {
    private static final long serialVersionUID = 1L;
    private JButton btnProcesar;
    private JScrollPane scpScroll;
    private JTextArea txtS;

    // Lanza la aplicación
    public static void main(String[] args) {
        try {
            UIManager

            .setLookAndFeel("com.sun.java.swing.plaf.windows.WindowsLookAndFeel");
        }
        catch (Throwable e) {
            e.printStackTrace();
        }
    }
}

```

```

        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Notas frame = new Notas();
                    frame.setVisible(true);
                }
                catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    // Crea la GUI
    public Notas() {
        setTitle("Notas");
        setBounds(100, 100, 450, 214);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        getContentPane().setLayout(null);

        btnProcesar = new JButton("Procesar");
        btnProcesar.addActionListener(this);
        btnProcesar.setBounds(173, 9, 89, 23);
        getContentPane().add(btnProcesar);

        scpScroll = new JScrollPane();
        scpScroll.setBounds(10, 44, 414, 120);
        getContentPane().add(scpScroll);

        txtS = new JTextArea();
        txtS.setFont(new Font("Monospaced", Font.PLAIN, 12));
        scpScroll.setViewportView(txtS);
    }

    // Direcciona eventos de tipo ActionEvent
    public void actionPerformed(ActionEvent arg0) {
        if (arg0.getSource() == btnProcesar) {
            actionPerformedBtnProcesar(arg0);
        }
    }

    // Procesa la pulsación del botón Procesar
    protected void actionPerformedBtnProcesar(ActionEvent arg0) {
        // Declara e inicializa variables
        int nota, capro = 0, cdesa = 0, suma = 0;
        double pro;

        // Proceso repetitivo
        for (int c = 0; c < 40; c++) {

            // Genera una nota
            nota = (int) (21 * Math.random());

            // Imprime la nota generada
            txtS.append(nota + "\n");

            // Suma la nota
            suma += nota;
        }
    }

```

```
        // Contabiliza la nota como aprobatoria o desaprobatoria
        if (nota >= 13)
            capro++;
        else
            cdesa++;
    }

    // Determina la nota promedio
    pro = suma * 1.0 / 25;

    // Imprime el reporte final
    txtS.append("Nota promedio          : " + pro + "\n");
    txtS.append("Cantidad de aprobados      : " + capro + "\n");
    txtS.append("Cantidad de desaprobados : " + cdesa + "\n");
}
}
```