TECNOLÓGICO NACIONAL DE MÉXICO

INSTITUTO TECNOLÓGICO DE OAXACA

ASIGNATURA: TOPICOS AVANZADOS DE PROGRAMACIÓN

CATEDRÁTICO: HERNANDEZ ABREGO ANAYANSI CRISTINA

ALUMNO: GARCÍA GARCÍA JOSÉ ÁNGEL

UNIDAD: 2

PRACTICA 2- Creación de una aplicación utilizando componentes propios y de biblioteca, utilizando un asistente de diseño.

OAXACA DE JÚAREZ, OAX, 10/MARZO/2020

Índice

BITACORA

9) Lo que realiza el código en el método **private void JButtoon1AcetionPerformed(java.awt.event.ActionEvent evt):**

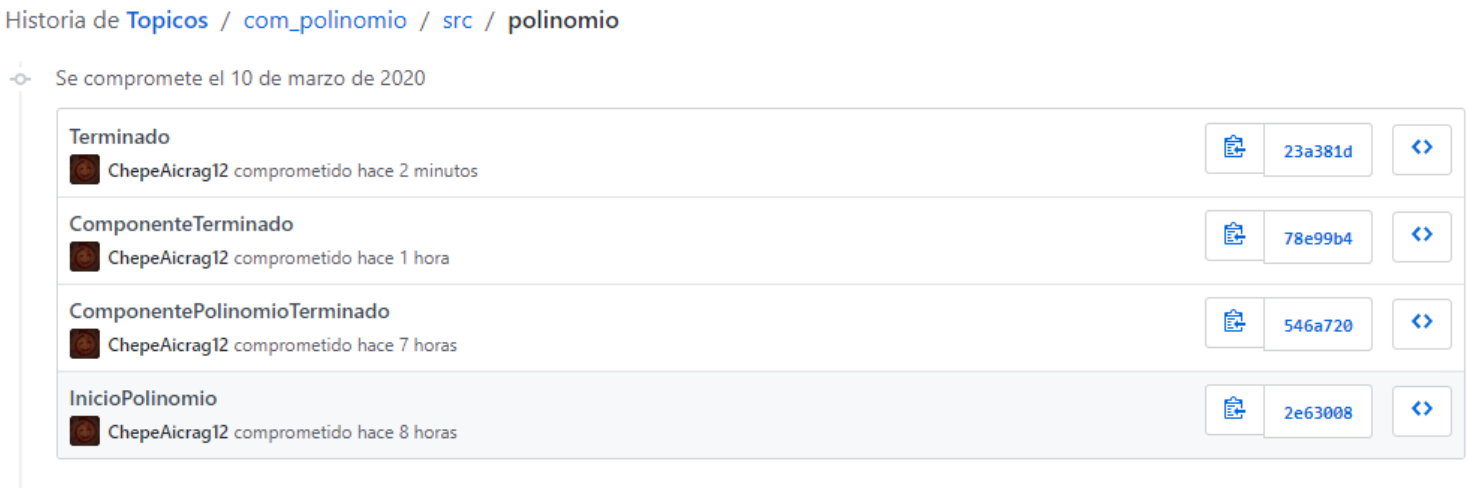
La acción generada por el botón “calcular” básicamente es que permite formar una cadena que contiene a las raíces pertenecientes a la función introducida. Es decir, ordena por cada caso a las raíces para mostrarla en el área de texto correspondiente. De igual forma para obtener dichas raíces, hace la ejecución de distintos métodos para poder calcularlas, pero obviamente, aquí es donde se hace uso del componente recientemente creado.

Lo que hace con Polinomio es fijar los valores de los coeficientes después de comprobar que estos realmente si son los coeficientes mínimos para una ecuación lineal. Por cada caso de numero de raíces, crea un arreglo de raíces de tamaño al grado de la misma. Por último, genera la cadena mencionada anteriormente dependiendo a los numero de raíces en la función.

11) Lo que realiza el método **private void creaGraficoActionPerfomed(java.awt.event.ActionEvent evt)**

Básicamente lo que se realiza en este método es crear la grafica de la función, esto con la ayuda de un complemento llamado JFreeChart. Lo fundamental para poder graficar es establecer los valores de inicio, final e incremento para que se pueda realizar dicho gráfico, de lo contrario no lo realiza. Se hace el uso de un total de 3 componentes de la librería importada. Cada uno de ellos funcionan para poder crear la gráfica.

Se hizo uso de GitHub para guardar cada cambio realizado durante la práctica. Aquí se muestra los cambios realizados de cada código. El primero es el del componente, el segundo corresponde a las pruebas del mismo.



Historia de **Topicos** / practica_5 / src / **aplicacion**

Se compromete el 10 de marzo de 2020

Terminado
ChepeAicrag12 comprometido hace 36 segundos
23a381d

AplicacionTerminada ...
ChepeAicrag12 comprometido hace 1 hora
c865563

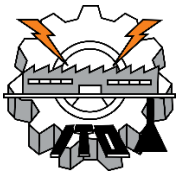
```
Se terminó la aplicación que hace uso del componente Polinomio
```

CÓDIGO FUENTE DEL COMPONENTE

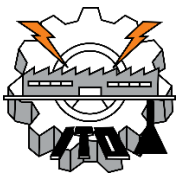
CÓDIGO

```
package polinomio;
import java.io.Serializable;
/**
 * @author Garcia Garcia Jose Angel
 */
public class Polinomio implements Serializable{
    protected double coeficiente[];
    protected static int  MAX_GRRADO = 3;
    protected static int MIN_GRADO = 1;
    protected int grado;
    public Polinomio(){
        coeficiente = new double[MAX_GRRADO+1];
        for (int i = 0; i <= MAX_GRRADO; i++) {
            coeficiente[i] = 0.0d;
        }
    }
    public Polinomio(double[] coeficiente) throws Exception{
        grado = coeficiente.length-1;
        if (coeficiente.length > 1) {
            this.coeficiente = coeficiente;
        }else{
            throw new Exception("Numero de coeficientes debe ser > 1");
        }
    }
    public void setCoeficiente(double coefs[]){
        grado = coefs.length-1;
        if(coefs.length > 1){
            for (int i = 0; i < coefs.length; i++) {
```

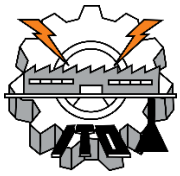
```
            coeficiente[i] = coefs[i];
        }}}
public int getGrado(){
   return grado;
}
public double[] vals(){
    double[] co = new double[coeficiente.length];
    for (int i = 0; i < coeficiente.length; i++) {
        co[i] = coeficiente[i];
    }
    return co;
}
public double getY(double x){
    double y = 0.0;
    for (int i = 0; i <= grado; i++) {
        y += coeficiente[i]*Math.pow(x, i);
    }
    return y;
}

public double getDerFX(double x){
    double dx = 0.0;
    for (int i = 1; i < grado; i++) {
        dx += i * coeficiente[i]*Math.pow(x, i-1);
    }
    return dx;
}

public String getPolinomio(){
    String polinomio = "";
```



```
  for ( int i = 0; i <= grado; i++ ) {
     if ( coeficiente[i] != 0 ){
        if ( i != 0 && coeficiente[i] > 0)
           polinomio += "+";
        if ( coeficiente[i] != 1 || i == 0 )
           polinomio +=  + coeficiente[i];
        if ( i > 0 )
           polinomio += "x";
        if ( i > 1 )
           polinomio += "^" + i;
     }
  }
    return polinomio;
}
public double[] getRaices() throws Exception{
    double x[] = null;
    // Para cuando es de grado 1
    if (grado == 1) {
        x = new double[grado];
        if (coeficiente[0] != 0) {
           x[0] = -1*coeficiente[0]/coeficiente[1];
        }else{
           throw new Exception("Coefciente debe X deber ser != 0");
        }
    }else if(grado == 2){
        x = new double[4];
        double a = coeficiente[2];
        double b = coeficiente[1];
        double c = coeficiente[0];
        double rad = Math.pow(b, 2.0) - 4*a*c;
```

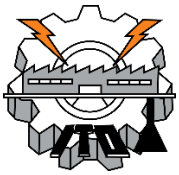


```
            if(rad >= 0){
                x[0] = (-b + Math.sqrt(rad))/(2*a);
                x[1] = (-b - Math.sqrt(rad))/(2*a);
            }else{ // Raices imaginarias
                x[0] = x[2] = -b/(2*a); // parte real
                x[1] = + Math.sqrt(Math.abs(rad))/(2*a); // Parte imaginaria + /
                x[3] = - Math.sqrt(Math.abs(rad))/(2*a); // Parte imagina -
            }
        }else if(grado > 2){
            // Utilizando el método de newton raphson
            final int MAX_ITERA = 100;
            final double PRECISION = 0.001;
            x = new double[grado];
            double error = 999999.0, x1 = 0;
            int i = 0, n = 0;
            for (n = 0; n < grado-2; n++) {
                i = 0;
                do {
                    x1 = x1 - getY(x1) / getDerFX(x1);
                    i++;
                } while (Math.abs(getY(x1)) > error && i == MAX_ITERA  );
                    x[n] = x1;
                }
        }
        return x;
    }
}
```

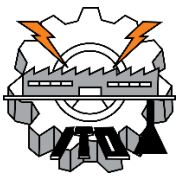


CÓDIGO FUENTE DE LA APLICACIÓN PRUEBA

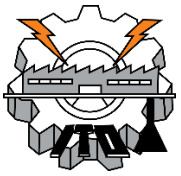
```
package aplicacion;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;
import org.jfree.chart.ChartFactory;
import org.jfree.chart.ChartFrame;
import org.jfree.chart.JFreeChart;
import org.jfree.chart.plot.PlotOrientation;
import org.jfree.data.xy.XYDataset;
import org.jfree.data.xy.XYSeries;
import org.jfree.data.xy.XYSeriesCollection;
import polinomio.Polinomio;
/**
 * @author Garcia Garcia Jose Angel
 */
public class Aplicacion extends javax.swing.JDialog {
    /**
     * Creates new form Aplicacion
     */
    private int MAX_NUM_COEF = 5;
    private Polinomio polinomio;
    private double coef[];
    private boolean calculo;
    private JFreeChart grafico;
    public Aplicacion(java.awt.Frame parent, boolean modal) {
        super(parent, modal);
        initComponents();
        coef = new double[MAX_NUM_COEF];
        calculo = false;
```



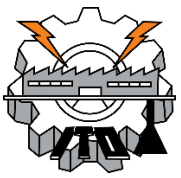
```
    a0.setValue(0.0);
    a1.setValue(0.0);
    a2.setValue(0.0);
    a3.setValue(0.0);
    vInicial.setValue(0.0);
    vFinal.setValue(0.0);
    vInc.setValue(0.0);

}
/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {
    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jLabel9 = new javax.swing.JLabel();
    jLabel10 = new javax.swing.JLabel();
    jLabel11 = new javax.swing.JLabel();
    jLabel12 = new javax.swing.JLabel();
    jLabel13 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    jLabel4 = new javax.swing.JLabel();
    jLabel5 = new javax.swing.JLabel();
    jLabel6 = new javax.swing.JLabel();
    jLabel7 = new javax.swing.JLabel();
    jLabel8 = new javax.swing.JLabel();
```



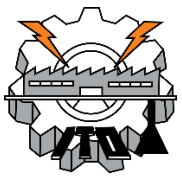
```
jLabel14 = new javax.swing.JLabel();
a0 = new javax.swing.JFormattedTextField();
a1 = new javax.swing.JFormattedTextField();
a2 = new javax.swing.JFormattedTextField();
a4 = new javax.swing.JFormattedTextField();
a3 = new javax.swing.JFormattedTextField();
a5 = new javax.swing.JFormattedTextField();
jButton1 = new javax.swing.JButton();
raices = new javax.swing.JTextField();
jLabel15 = new javax.swing.JLabel();
jLabel16 = new javax.swing.JLabel();
jLabel17 = new javax.swing.JLabel();
jLabel18 = new javax.swing.JLabel();
jButton2 = new javax.swing.JButton();
vInicial = new javax.swing.JFormattedTextField();
vFinal = new javax.swing.JFormattedTextField();
vInc = new javax.swing.JFormattedTextField();

setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
jLabel1.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
jLabel1.setText("Calculo y graficación de Polinomios");
jLabel2.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel2.setText("De valores a coeficientes");
jLabel9.setFont(new java.awt.Font("Tahoma", 0, 20)); // NOI18N
jLabel9.setText("a");
jLabel10.setFont(new java.awt.Font("Tahoma", 0, 20)); // NOI18N
jLabel10.setText("a");
jLabel11.setFont(new java.awt.Font("Tahoma", 0, 20)); // NOI18N
jLabel11.setText("a");
```



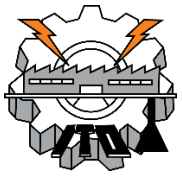
```
jLabel12.setFont(new java.awt.Font("Tahoma", 0, 20)); // NOI18N
jLabel12.setText("a");
jLabel13.setFont(new java.awt.Font("Tahoma", 0, 20)); // NOI18N
jLabel13.setText("a");
jLabel3.setText("0");
jLabel4.setText("1");
jLabel5.setText("2");
jLabel6.setText("3");
jLabel7.setText("4");
jLabel8.setText("5")
jLabel14.setFont(new java.awt.Font("Tahoma", 0, 20)); // NOI18N
jLabel14.setText("a");
a0.setText("0.0");
a0.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        a0ActionPerformed(evt); }});
a1.setText("0.0");
a1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        a1ActionPerformed(evt);
    }});
a2.setText("0.0");
a2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        a2ActionPerformed(evt);
    }});
a4.setText("0.0");
a3.setText("0.0");
a3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
```



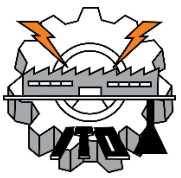
```
        a3ActionPerformed(evt);
    }});
a5.setText("0.0");
jButton1.setText("Calcular");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }});
raices.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        raicesActionPerformed(evt);
    }});
jLabel15.setText("GRAFICAR:");
jLabel16.setText("Valor Inicial");
jLabel17.setText("Valor Final");
jLabel18.setText("Incremento");
jButton2.setText("CREAR GRAFICO");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }});
javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addContainerGap()
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```



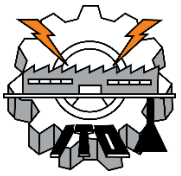
```
                .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 294,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGroup(layout.createSequentialGroup()
                    .addGap(10, 10, 10)
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                        .addGroup(layout.createSequentialGroup()
                            .addComponent(jButton1)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                            .addComponent(raices, javax.swing.GroupLayout.PREFERRED_SIZE, 293,
javax.swing.GroupLayout.PREFERRED_SIZE))
                        .addComponent(jLabel2)
                        .addGroup(layout.createSequentialGroup()
                            .addComponent(jLabel15)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                            .addComponent(jLabel16)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                            .addComponent(vInicial, javax.swing.GroupLayout.PREFERRED_SIZE, 34,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                            .addComponent(jLabel17)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                            .addComponent(vFinal, javax.swing.GroupLayout.PREFERRED_SIZE, 34,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                            .addComponent(jLabel18)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                            .addComponent(vInc, javax.swing.GroupLayout.PREFERRED_SIZE, 34,
javax.swing.GroupLayout.PREFERRED_SIZE))
                        .addComponent(jButton2)))))
```



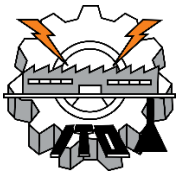
```
                .addGroup(layout.createSequentialGroup()
                    .addGap(88, 88, 88)
                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING,
false)
                        .addGroup(layout.createSequentialGroup()
                            .addComponent(jLabel9)
                            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                            .addComponent(jLabel3)
                            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                            .addComponent(a0, javax.swing.GroupLayout.PREFERRED_SIZE, 43,
javax.swing.GroupLayout.PREFERRED_SIZE))
                        .addGroup(layout.createSequentialGroup()
                            .addComponent(jLabel10)
                            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                            .addComponent(jLabel4)
                            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                            .addComponent(a1))
                        .addGroup(layout.createSequentialGroup()
                            .addComponent(jLabel11)
                            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                            .addComponent(jLabel5)
                            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                            .addComponent(a2, javax.swing.GroupLayout.PREFERRED_SIZE, 42,
javax.swing.GroupLayout.PREFERRED_SIZE))
                        .addGroup(layout.createSequentialGroup()
                            .addComponent(jLabel12)
                            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                            .addComponent(jLabel6)
                            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                            .addComponent(a3, javax.swing.GroupLayout.PREFERRED_SIZE, 42,
javax.swing.GroupLayout.PREFERRED_SIZE))
```



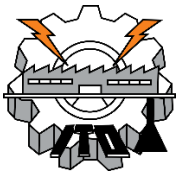
```
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
                        .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
layout.createSequentialGroup()
                            .addComponent(jLabel14)
                            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                            .addComponent(jLabel8)
                            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                            .addComponent(a5))
                        .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
layout.createSequentialGroup()
                            .addComponent(jLabel13)
                            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                            .addComponent(jLabel7)
                            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                            .addComponent(a4, javax.swing.GroupLayout.PREFERRED_SIZE, 42,
javax.swing.GroupLayout.PREFERRED_SIZE))))))
            .addContainerGap(74, Short.MAX_VALUE))
        );
        layout.setVerticalGroup(
           layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
           .addGroup(layout.createSequentialGroup()
              .addGap(23, 23, 23)
              .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 26,
javax.swing.GroupLayout.PREFERRED_SIZE)
              .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
              .addComponent(jLabel2)
              .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
              .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                    .addComponent(jLabel9)
                    .addComponent(a0, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
```



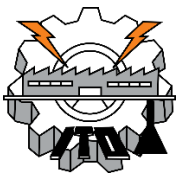
```
                .addComponent(jLabel3, javax.swing.GroupLayout.Alignment.TRAILING))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                        .addComponent(jLabel10)
                        .addComponent(a1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                    .addGap(10, 10, 10))
                .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
                    .addComponent(jLabel4)
                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)))
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                        .addComponent(jLabel11)
                        .addComponent(a2, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                    .addComponent(jLabel12))
                .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
                    .addComponent(jLabel5)
                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                        .addComponent(a3, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addComponent(jLabel6))))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
```



```
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(jLabel13)
                    .addComponent(a4, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                .addComponent(jLabel7, javax.swing.GroupLayout.Alignment.TRAILING))
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                        .addComponent(jLabel14)
                        .addComponent(a5, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)))
                .addGroup(layout.createSequentialGroup()
                    .addGap(24, 24, 24)
                    .addComponent(jLabel8)))
            .addGap(35, 35, 35)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jButton1)
                .addComponent(raices, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(18, 18, 18)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel15)
                .addComponent(jLabel16)
                .addComponent(jLabel17)
                .addComponent(jLabel18)
                .addComponent(vInicial, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(vFinal, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
```

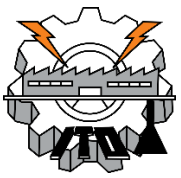


```
                .addComponent(vInc, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jButton2)
            .addContainerGap())
        );
        pack();
    }// </editor-fold>
    private double valor(Number e){
        return (double)(e);
    }
    private void a0ActionPerformed(java.awt.event.ActionEvent evt) {
      coef[0] = valor((Number)a0.getValue());
      if(coef[0] == 0.0){
          a0.requestFocus();
          JOptionPane.showMessageDialog(this, "El valor del coeficiente tiene que ser != 0");
      }else{
          a1.requestFocus();
      }
    }
    private void a1ActionPerformed(java.awt.event.ActionEvent evt) {
        coef[1] = valor((Number)a1.getValue());
      if(coef[1] == 0.0){
          a1.requestFocus();
          JOptionPane.showMessageDialog(this, "El valor del coeficiente tiene que ser != 0");
      }else{
          a2.requestFocus();
      }
    }
```

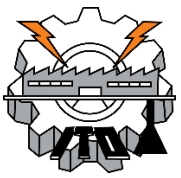


```
private void raicesActionPerformed(java.awt.event.ActionEvent evt) {}
private void a2ActionPerformed(java.awt.event.ActionEvent evt) {
   coef[2] = valor((Number)a2.getValue());
   if(coef[2] == 0.0){
       a2.requestFocus();
       JOptionPane.showMessageDialog(this, "El valor del coeficiente tiene que ser != 0");
   }else{
       a3.requestFocus();
   }  }
private void a3ActionPerformed(java.awt.event.ActionEvent evt) {
  coef[3] = valor((Number)a3.getValue());
   if(coef[3] == 0.0){
       a3.requestFocus();
       JOptionPane.showMessageDialog(this, "El valor del coeficiente tiene que ser != 0");
   }else{
       a4.requestFocus();
   } }
private boolean verificaValores(){
    try {
        double a  = coef[0];
        double b = coef[1];
        if(a == 0 && b == 0)
            return false;
    } catch (Exception e) {
        return false;
    }
    return true;
}
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
  String tRaices = ""; // Texto a mostrarde las raices
```

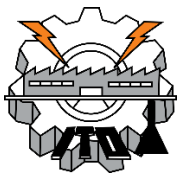
```
double vRaices[] = null; // Valores de las raices
double coefs[] = null; // Valores de los coeficientes
int nRaices = 0; // numero de raices obtenidas
int nCoef = 0; // numerode coeficientes
polinomio = new Polinomio();
// valida que sean los coeficientes minimos para una ecuacion lineal
calculo = verificaValores();
if(calculo){
    nCoef = 2;
    if(coef[2] != 0){
        nCoef++;
        vRaices = new double[4];
    }if(coef[3] != 0){
        vRaices= new double[3];
        nCoef++;
    }
    coefs = new double[nCoef];
    for (int c = 0; c < nCoef; c++) {
        coefs[c] = coef[c];
    }
    // se fijan los valores de los coeficientes
    polinomio.setCoeficiente(coefs);
    try {
        vRaices = polinomio.getRaices();
        nRaices = polinomio.getRaices().length;
    } catch (Exception e) {
        Logger.getLogger(Aplicacion.class.getName()).log(Level.SEVERE,null,e);
    }if(polinomio.getGrado() == 1){
        tRaices = String.format(" X = %6.4f", vRaices[0]);
    }if(polinomio.getGrado() == 2){
```



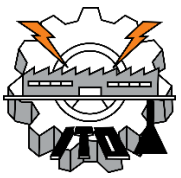
```
            if(nRaices == 2)
            tRaices = String.format(" X0 = %6.4f , x1 = %6.4f ", vRaices[0],vRaices[1]);
            else
            tRaices = String.format(" X0 = %6.4f +%6.4f i    X1 = %6.4f - %6.4f i ",
vRaices[0],vRaices[1],vRaices[2],vRaices[1]);
          }else
            if(polinomio.getGrado() > 2)
               for (int r = 0; r < nRaices; r++) {
                  tRaices += String.format("X[%d]=%6.4f,", r,vRaices[r]);
         }
          tRaices = tRaices.substring(0, tRaices.length()-1);
          if(tRaices != ""){
            raices.setText(tRaices);
            vInicial.setEditable(true);
            vFinal.setEditable(true);
            vInc.setEditable(true);
          }
     }else{
        JOptionPane.showMessageDialog(this, "No tiene valores para realizar el calculo");
     }
   }
   private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
     if(valor((Number)vInicial.getValue()) != 0 && valor((Number)vFinal.getValue()) != 0 &&
valor((Number)vInc.getValue()) != 0){
        XYSeries serieXY = new XYSeries("y=f(x)");
        double vf = valor((Number)vFinal.getValue());
        double vi = valor((Number)vInicial.getValue());
        double inc = valor((Number)vInc.getValue());
        while(vi < vf){
           serieXY.add(vi,polinomio.getY(vi));
           vi += inc;
```



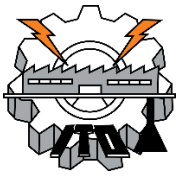
```
        }
        XYDataset datosXY = new XYSeriesCollection(serieXY);
        grafico = ChartFactory.createXYLineChart("y="+polinomio.getPolinomio(), "X", "Y", datosXY,
                PlotOrientation.VERTICAL,true,true,false);
        ChartFrame fgrafico = new ChartFrame("", grafico);
        fgrafico.setVisible(true);
        fgrafico.setEnabled(true);
        fgrafico.setBounds(100, 100, 500, 500);
     }
    }
    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
         * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Aplicacion.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
```



```
        } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Aplicacion.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(Aplicacion.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(Aplicacion.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
        }
        //</editor-fold>
        /* Create and display the dialog */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                Aplicacion dialog = new Aplicacion(new javax.swing.JFrame(), true);
                dialog.addWindowListener(new java.awt.event.WindowAdapter() {
                    @Override
                    public void windowClosing(java.awt.event.WindowEvent e) {
                        System.exit(0);
                    }
                });
                dialog.setVisible(true);
            }
        });
    }
    // Variables declaration - do not modify
    private javax.swing.JFormattedTextField a0;
    private javax.swing.JFormattedTextField a1;
    private javax.swing.JFormattedTextField a2;
    private javax.swing.JFormattedTextField a3;
```



```
private javax.swing.JFormattedTextField a4;
private javax.swing.JFormattedTextField a5;
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel10;
private javax.swing.JLabel jLabel11;
private javax.swing.JLabel jLabel12;
private javax.swing.JLabel jLabel13;
private javax.swing.JLabel jLabel14;
private javax.swing.JLabel jLabel15;
private javax.swing.JLabel jLabel16;
private javax.swing.JLabel jLabel17;
private javax.swing.JLabel jLabel18;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;
private javax.swing.JLabel jLabel7;
private javax.swing.JLabel jLabel8;
private javax.swing.JLabel jLabel9;
private javax.swing.JTextField raices;
private javax.swing.JFormattedTextField vFinal;
private javax.swing.JFormattedTextField vInc;
private javax.swing.JFormattedTextField vInicial;
// End of variables declaration          }
```



PRUEBA DEL COMPONENTE “POLINOMIO”

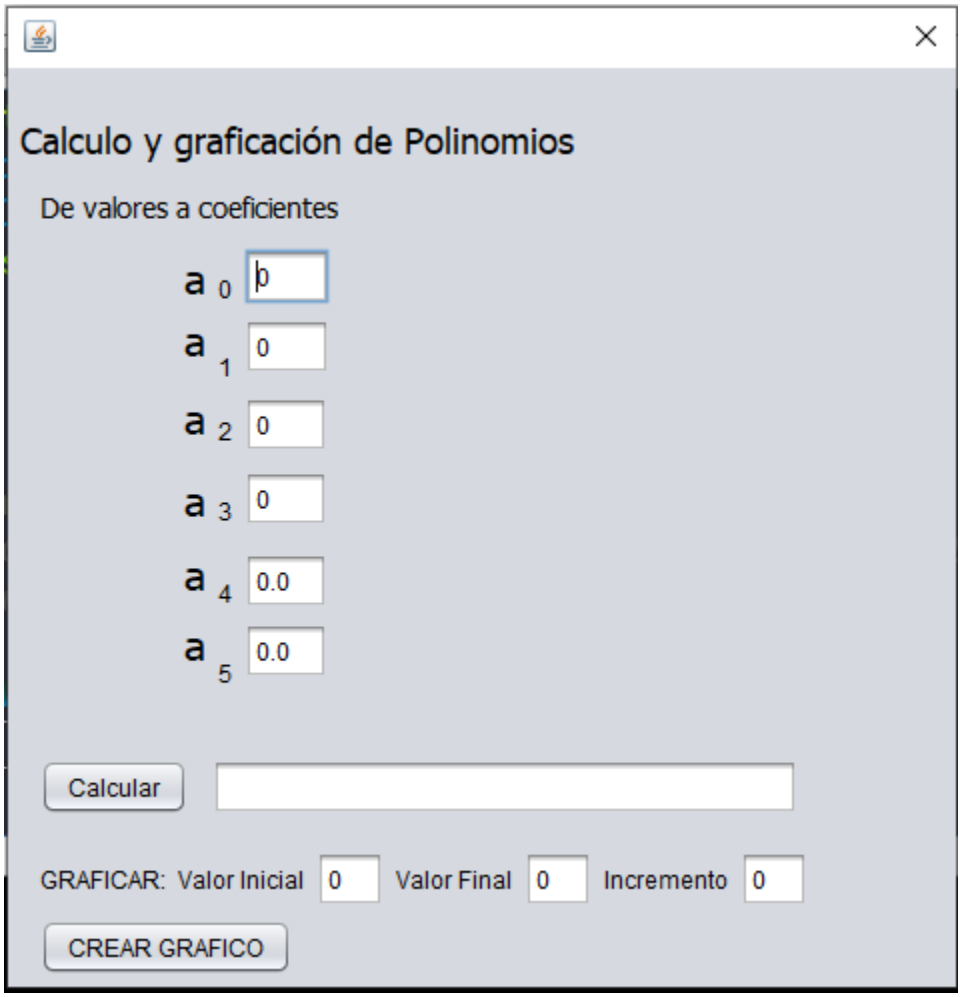
Se prueba el componente Polinomio. Esto desde una clase externa, pero en el mismo paquete de la clase del componente.

```
10      *
11      * @author Garcia Garcia Jose Angel
12      */
13     public class Prueba {
14         public static void main(String[] args) throws Exception {
15             double[] a = {1,5,2,10};
16             Polinomio p = new Polinomio(a);
17             double[] raices = p.getRaices();
18             System.out.println(p.getPolinomio());
19             for (int i = 0; i < raices.length; i++) {
20                 System.err.println(" La raiz " +  i + " es " + raices[i]);
21             }
22
23         }
24     }
```

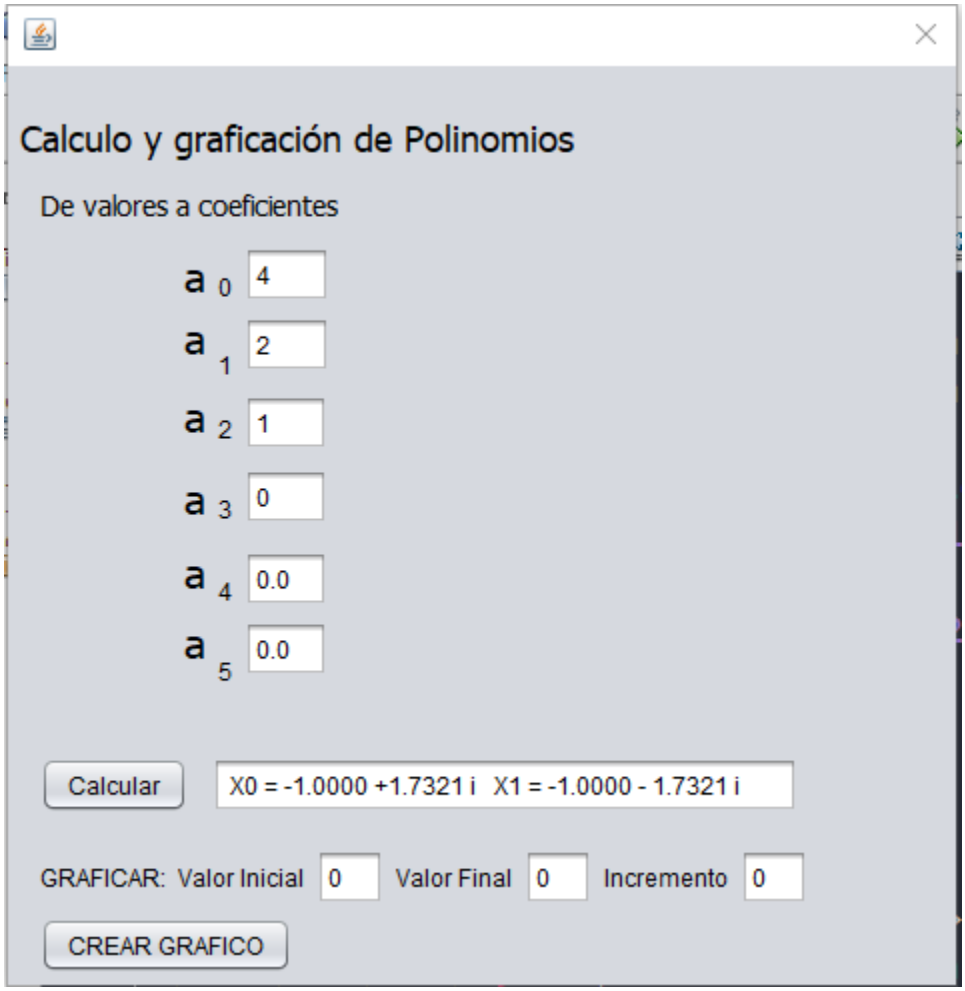
```
Output - com_polinomio (run)
run:
1.0+5.0x+2.0x^2+10.0x^3
 La raiz 0 es -0.2
 La raiz 1 es 0.0
 La raiz 2 es 0.0
BUILD SUCCESSFUL (total time: 0 seconds)
```

PRUEBA DE APLICACIÓN CON COMPONENTE

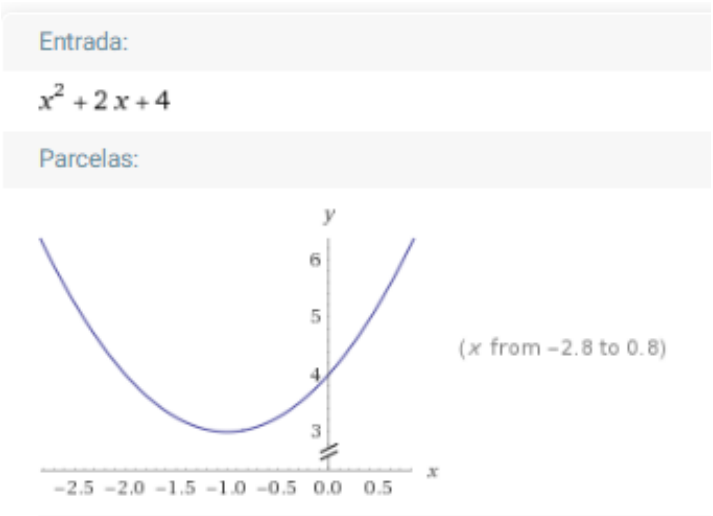
La interfaz que se muestra al iniciar la ejecución es la siguiente.



Rellenando con datos aleatorios para probar el programa.



Comprobando con un software especializado para las soluciones de problemas matemáticos.



$$x(x+2)+4$$

$$(x+1)^2+3$$

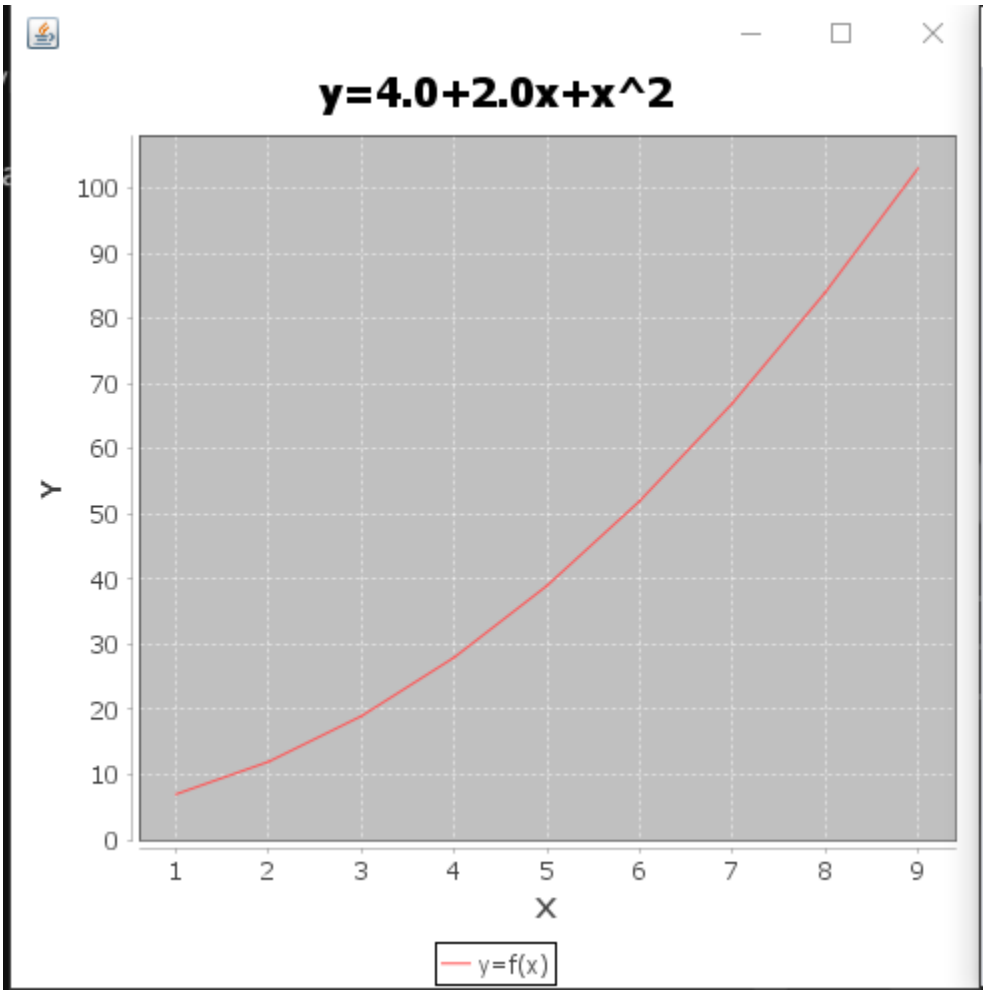
$$\left(-ix+\sqrt{3}-i\right)\left(ix+\sqrt{3}+i\right)$$

Raíces complejas:

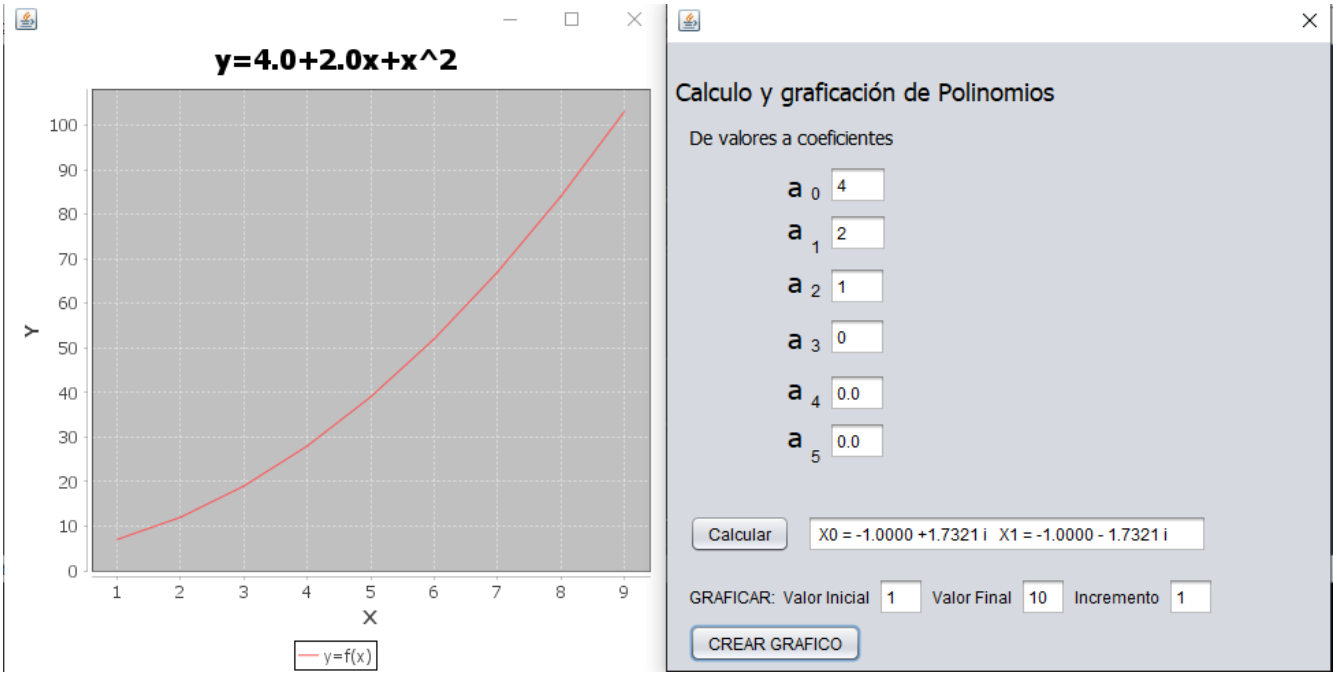
$$x=-1-i\sqrt{3}$$

$$x=-1+i\sqrt{3}$$

Graficando la función, obtendríamos el siguiente resultado.



De forma general, la aplicación se visualizaría así. Entonces comprobamos que funciona de manera correcta.



Prueba de que se está haciendo de las librerías solicitadas. En especial que se hace uso del componente Polinomio.