

Resum

Aquest annex conté el codi sencer de l'aplicació Conveyor Optimizer escrit en llenguatge de programació Java.

El codi està constantment comentat amb anotacions per tal que un lector amb els suficients coneixements de programació pugui entendre el que s'està implementat i les operacions que realitza l'algorisme en cada moment.

Sumari

RESUM	<hr/> 1
SUMARI	<hr/> 3
1. CODI DE L'APLICACIÓ	<hr/> 5

1. Codi de l'aplicació

A continuació, es mostra el codi sencer de l'aplicació en llenguatge Java.

El codi està comentat amb anotacions constantment. A mesura que avança l'algorisme, es va comentant per anar seguint les operacions que realitza i per entendre el propi codi.

```

package conveyoroptimizer;

import java.awt.Graphics;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;
import java.lang.Math;

public class Interface extends javax.swing.JFrame {

    //final int JPanel7medio = 322;
    /*VARIABLES*/
    /*Variables de idioma*/
    int idioma = 1; // 1: Español, 2: Català, 3: English, 4: Deutsch
    /*Mensajes emergentes*/
    String ComboBox1 = "Última";
    String ComboBox2 = "Penúltima";
    String mensaje1 = "Todos los valores han sido leídos correctamente";
    String mensaje2 = "Por favor, primero seleccionar un archivo válido en Tipo de cinta";
    String mensaje3 = "La fuerza máxima por tope debe ser mayor o igual que la fuerza ejercida por paleta";
    String mensaje4 = "El paso de paleta introducido es menor que el paso mínimo de este tipo de cinta";
    String mensaje5 = "La distancia de carga introducida es menor que la distancia de carga mínima de este tipo de cinta";
    String mensaje6 = "La distancia de descarga introducida es menor que la distancia de descarga mínima de este tipo de cinta";
    String mensaje7 = "El sensor de carga está a menos de medio paso de paleta de la posición de carga";
    String mensaje8 = "El sensor de descarga está a menos de medio paso de paleta del fin del giro de vuelta";
    String mensaje9 = "La distancia entre la posición de descarga y la posición de espera a la descarga es inferior a <paso + delta tope> o a <K4 + paso/2 + delta tope>";
    String mensaje10 = "La distancia entre el inicio del giro de ida y la posició de espera a la carga es inferior al paso de paleta + la constante del tope";
    String mensaje11 = "La distancia entre ejes introducida es demasiado pequeña en la ida";
    String mensaje12 = "La distancia entre ejes introducida es demasiado pequeña en la vuelta";
    String mensaje13 = "El tiempo de ciclo debe ser mayor a la suma del tiempo de carga y de descarga";
    String mensaje14 = "Todos los parámetros son compatibles con este tipo de cinta";
    String mensaje15 = "Por favor, primero introducir y validar los parámetros";
    String mensaje16 = "Por favor, primero obtener los Resultados geométricos";
    String mensaje17 = "Velocidad demasiado pequeña. No se ha obtenido una solución.";
    String mensaje18 = "La velocidad introducida es inferior a la velocidad mínima";
    String mensaje19 = "La velocidad introducida es superior a la velocidad máxima";
    String mensaje20 = "Pulmón vacío:";
    String mensaje21 = "Pulmón vacío: Sí";
    String mensaje22 = "Pulmón vacío: No";
    String mensaje23 = "El número de paletas deseado no puede ser inferior a 1";
    String mensaje24 = "El número de paletas deseado no puede ser superior al número de paletas máximo";
    String mensaje25 = "Se ha impuesto velocidad mínima";
    String mensaje26 = "Número de paletas demasiado pequeño. La velocidad excede la máxima.";
    String mensaje27 = "Hay más paletas de las necesarias con la velocidad obtenida";
    /*Variables pestaña "Tipo de cinta"*/
    double K1; /*Fuerza max aquanta cada tope*/
    double D; /*Diámetro de giro*/
    double dCar_min; /*Distancia de carga minima*/
    double dDesc_min; /*Distancia de descarga minima*/
    double pas_min; /*Paso minimo*/
    double delt atop; /*Delta topall*/
    double v_min; /*Velocidad cinta minima*/
    double v_max; /*Velocidad cinta maxima*/
    double dG_ida; /*Distancia despues del giro de ida*/
    double dG_v; /*Distancia despues del giro de vuelta*/
    double dEG_ida; /*Distancia antes del giro de ida*/
    double dEG_v; /*Distancia antes del giro de vuelta*/
    double K5; /*Constante del tope de espera a la carga*/
    boolean archivo_leido = false; /*Boleano para indicar que se ha leído correctamente el archivo*/
    /*Variables pestaña "Parámetros"*/
}

```

```

double L; /*Distancia entre ejes*/
double dCar; /*Distancia entre fin giro ida y posicion carga*/
double dDesc; /*Distancia entre inicio giro vuelta y posicion descarga*/
double dSCar; /*Distancia entre posicion carga y sensor carga*/
double dSDesc; /*Distancia entre fin giro vuelta y sensor descarga*/
double dT_ida01; /*Distancia entre posicion descarga y posicion espera descarga*/
double dT_v01; /*Distancia entre inicio giro ida y posicion espera carga*/
double K2; /*Fuerza que hace cada paleta*/
double pas; /*Paso de paleta*/
double K3; /*Espacio carga*/
double K4; /*Espacio descarga*/
double tCiclo; /*Tiempo de ciclo*/
double tCarga; /*Tiempo de carga*/
double tDescarga; /*Tiempo de descarga*/
double tNeto; /*Tiempo neto resultante*/
boolean error_validar = true; /*Boleano para indicar que NO hay ningun error al validar*/
/*Variables pestaña "Resultados geometricos"*/
boolean ResGeom = false; /*Boleano para indicar que se han obtenido resultados geométricos */
/*IDA*/
int Nmax_ida; /*Numero paletas maximas en la cola de ida*/
int Ntop_ida; /*Numero de topes en la cola de ida*/
int tamgrups_ida; /*Tamaño grupos en la cola de ida*/
String coordenadasTop_ida; /*Texto con las coordenadas de los topes de ida referenciadas al eje de vuelta*/
int NSens_ida; /*Numero de sensores totales en lado de la ida*/
int Nultimgrup_ida; /*Numero paletas del ultimo grupo (i.e. que aguanta el ultimo tope)*/
boolean penultimapal_ida = false; /*Boleano true si sensor de tope va en la penultima paleta en la ida (por defecto va en la ultima)*/
String coordenadasSens_ida; /*Texto con las distancias de los sensores de ida referenciadas al eje de vuelta
int Narriba; /*Numero de paletas total que hay arriba
/*VUELTA*/
int Nmax_v; /*Numero paletas maximas en la cola de vuelta
int Ntop_v; /*Numero de topes en la cola de vuelta
int tamgrups_v; /*Tamaño grupos en la cola de vuelta
String coordenadasTop_v; /*Texto con las distancias de los topes de vuelta referenciadas al eje de ida
int NSens_v; /*Numero de sensores totales en lado de la vuelta
int Nultimgrup_v; /* Numero paletas del ultimo grupo (i.e. que aguanta el ultimo tope
boolean penultimapal_v = false; // Boleano true si sensor de tope va en la penultima paleta en la vuelta (por defecto va en la ultima)
String coordenadastSens_v; // Texto con las distancias de los sensores de ida referenciadas al eje de ida
int Nabajo; // Numero de paletas total que hay abajo
/*TOTAL*/
int Nmax_TOT; // Numero de paletas maximas que caben en el transportador
int Ntop_TOT; // Numero de topes totales
int NSens_TOT; // Numero de sensores totales
/*FIN VARIABLES*/
/*Crear vector de prueba*/

/**
 * Creates new form Interface
 */
public Interface() {
    initComponents();
    /* Que la intefaz aparezca en medio de la pantalla
}

/**
 * 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() {

```

```
jTabbedPane1 = new javax.swing.JTabbedPane();
jPanel1 = new javax.swing.JPanel();
jLabel1 = new javax.swing.JLabel();
jLabel2 = new javax.swing.JLabel();
btnCatala = new javax.swing.JButton();
btnEspañol = new javax.swing.JButton();
btnEnglish = new javax.swing.JButton();
btnDeutsch = new javax.swing.JButton();
jLabel3 = new javax.swing.JLabel();
jLabel4 = new javax.swing.JLabel();
jLabel5 = new javax.swing.JLabel();
jLabel6 = new javax.swing.JLabel();
jLabel163 = new javax.swing.JLabel();
jPanel2 = new javax.swing.JPanel();
jLabel162 = new javax.swing.JLabel();
jLabel165 = new javax.swing.JLabel();
jLabel166 = new javax.swing.JLabel();
jLabel167 = new javax.swing.JLabel();
jLabel168 = new javax.swing.JLabel();
jLabel169 = new javax.swing.JLabel();
jPanel3 = new javax.swing.JPanel();
jLabel7 = new javax.swing.JLabel();
btnLeerArchivoCinta = new javax.swing.JButton();
jTextField1 = new javax.swing.JTextField();
jLabel8 = new javax.swing.JLabel();
jTextField2 = new javax.swing.JTextField();
jTextField3 = new javax.swing.JTextField();
jLabel9 = new javax.swing.JLabel();
jTextField4 = new javax.swing.JTextField();
jLabel10 = new javax.swing.JLabel();
jTextField5 = new javax.swing.JTextField();
jLabel11 = new javax.swing.JLabel();
jTextField6 = new javax.swing.JTextField();
jLabel12 = new javax.swing.JLabel();
jTextField7 = new javax.swing.JTextField();
jLabel13 = new javax.swing.JLabel();
jTextField8 = new javax.swing.JTextField();
jLabel14 = new javax.swing.JLabel();
jLabel15 = new javax.swing.JLabel();
jTextField9 = new javax.swing.JTextField();
jLabel16 = new javax.swing.JLabel();
jLabel17 = new javax.swing.JLabel();
jLabel18 = new javax.swing.JLabel();
jLabel19 = new javax.swing.JLabel();
jLabel20 = new javax.swing.JLabel();
jLabel21 = new javax.swing.JLabel();
jLabel22 = new javax.swing.JLabel();
jLabel23 = new javax.swing.JLabel();
jLabel24 = new javax.swing.JLabel();
jLabel61 = new javax.swing.JLabel();
jTextField25 = new javax.swing.JTextField();
jLabel62 = new javax.swing.JLabel();
jLabel63 = new javax.swing.JLabel();
jTextField26 = new javax.swing.JTextField();
jLabel64 = new javax.swing.JLabel();
jLabel65 = new javax.swing.JLabel();
jTextField27 = new javax.swing.JTextField();
jLabel66 = new javax.swing.JLabel();
jLabel67 = new javax.swing.JLabel();
jTextField28 = new javax.swing.JTextField();
jLabel68 = new javax.swing.JLabel();
jLabel83 = new javax.swing.JLabel();
jTextField35 = new javax.swing.JTextField();
jLabel84 = new javax.swing.JLabel();
```

```
jLabel161 = new javax.swing.JLabel();
jPanel4 = new javax.swing.JPanel();
jLabel25 = new javax.swing.JLabel();
jLabel26 = new javax.swing.JLabel();
jTextField10 = new javax.swing.JTextField();
jLabel27 = new javax.swing.JLabel();
jLabel28 = new javax.swing.JLabel();
jLabel29 = new javax.swing.JLabel();
jTextField11 = new javax.swing.JTextField();
jLabel30 = new javax.swing.JLabel();
jLabel31 = new javax.swing.JLabel();
jLabel32 = new javax.swing.JLabel();
jTextField13 = new javax.swing.JTextField();
jTextField12 = new javax.swing.JTextField();
jLabel33 = new javax.swing.JLabel();
jLabel34 = new javax.swing.JLabel();
jLabel35 = new javax.swing.JLabel();
jTextField14 = new javax.swing.JTextField();
jLabel36 = new javax.swing.JLabel();
jLabel37 = new javax.swing.JLabel();
jTextField15 = new javax.swing.JTextField();
jLabel38 = new javax.swing.JLabel();
jLabel39 = new javax.swing.JLabel();
jTextField16 = new javax.swing.JTextField();
jLabel40 = new javax.swing.JLabel();
jLabel41 = new javax.swing.JLabel();
jLabel42 = new javax.swing.JLabel();
jTextField17 = new javax.swing.JTextField();
jLabel43 = new javax.swing.JLabel();
jLabel44 = new javax.swing.JLabel();
jLabel45 = new javax.swing.JLabel();
jLabel46 = new javax.swing.JLabel();
jTextField18 = new javax.swing.JTextField();
jTextField19 = new javax.swing.JTextField();
jTextField20 = new javax.swing.JTextField();
jLabel47 = new javax.swing.JLabel();
jLabel48 = new javax.swing.JLabel();
jLabel49 = new javax.swing.JLabel();
jLabel50 = new javax.swing.JLabel();
jLabel51 = new javax.swing.JLabel();
jLabel52 = new javax.swing.JLabel();
jLabel53 = new javax.swing.JLabel();
jLabel54 = new javax.swing.JLabel();
jLabel55 = new javax.swing.JLabel();
jTextField21 = new javax.swing.JTextField();
jTextField22 = new javax.swing.JTextField();
jTextField23 = new javax.swing.JTextField();
jTextField24 = new javax.swing.JTextField();
jLabel56 = new javax.swing.JLabel();
jLabel57 = new javax.swing.JLabel();
jLabel58 = new javax.swing.JLabel();
jLabel59 = new javax.swing.JLabel();
jLabel60 = new javax.swing.JLabel();
chkPulmonVacio = new javax.swing.JCheckBox();
btnValidarParametros = new javax.swing.JButton();
jLabel85 = new javax.swing.JLabel();
jTextField36 = new javax.swing.JTextField();
jLabel86 = new javax.swing.JLabel();
jLabel155 = new javax.swing.JLabel();
jPanel5 = new javax.swing.JPanel();
jLabel69 = new javax.swing.JLabel();
jLabel70 = new javax.swing.JLabel();
jLabel71 = new javax.swing.JLabel();
jLabel72 = new javax.swing.JLabel();
```

```
jLabel73 = new javax.swing.JLabel();
jLabel74 = new javax.swing.JLabel();
jTextField29 = new javax.swing.JTextField();
jTextField30 = new javax.swing.JTextField();
jTextField31 = new javax.swing.JTextField();
jTextField32 = new javax.swing.JTextField();
jTextField33 = new javax.swing.JTextField();
jLabel96 = new javax.swing.JLabel();
jLabel97 = new javax.swing.JLabel();
jLabel88 = new javax.swing.JLabel();
jLabel89 = new javax.swing.JLabel();
jLabel90 = new javax.swing.JLabel();
jTextField37 = new javax.swing.JTextField();
jTextField38 = new javax.swing.JTextField();
jTextField39 = new javax.swing.JTextField();
jLabel91 = new javax.swing.JLabel();
btnResultadosGeometricos = new javax.swing.JButton();
jLabel81 = new javax.swing.JLabel();
jTextField34 = new javax.swing.JTextField();
chkRepartirTopes_ida = new javax.swing.JCheckBox();
ComboBxPenultim_ida = new javax.swing.JComboBox();
jLabel87 = new javax.swing.JLabel();
jLabel95 = new javax.swing.JLabel();
jTextField47 = new javax.swing.JTextField();
chkRepartirTopes_v = new javax.swing.JCheckBox();
jLabel82 = new javax.swing.JLabel();
jTextField40 = new javax.swing.JTextField();
jTextField48 = new javax.swing.JTextField();
jLabel98 = new javax.swing.JLabel();
jLabel75 = new javax.swing.JLabel();
jLabel76 = new javax.swing.JLabel();
jLabel77 = new javax.swing.JLabel();
jLabel99 = new javax.swing.JLabel();
jLabel78 = new javax.swing.JLabel();
jTextField41 = new javax.swing.JTextField();
jTextField42 = new javax.swing.JTextField();
jLabel79 = new javax.swing.JLabel();
ComboBxPenultim_v = new javax.swing.JComboBox();
jLabel100 = new javax.swing.JLabel();
jTextField43 = new javax.swing.JTextField();
jTextField44 = new javax.swing.JTextField();
jLabel92 = new javax.swing.JLabel();
jTextField45 = new javax.swing.JTextField();
jLabel80 = new javax.swing.JLabel();
jTextField50 = new javax.swing.JTextField();
jLabel93 = new javax.swing.JLabel();
jLabel94 = new javax.swing.JLabel();
jTextField51 = new javax.swing.JTextField();
jLabel102 = new javax.swing.JLabel();
jLabel101 = new javax.swing.JLabel();
jLabel103 = new javax.swing.JLabel();
jLabel104 = new javax.swing.JLabel();
jLabel105 = new javax.swing.JLabel();
jTextField52 = new javax.swing.JTextField();
jLabel106 = new javax.swing.JLabel();
jLabel107 = new javax.swing.JLabel();
jLabel108 = new javax.swing.JLabel();
jLabel109 = new javax.swing.JLabel();
jLabel110 = new javax.swing.JLabel();
jLabel111 = new javax.swing.JLabel();
jLabel156 = new javax.swing.JLabel();
jPanel6 = new javax.swing.JPanel();
btnVN = new javax.swing.JButton();
jLabel127 = new javax.swing.JLabel();
```

```
jLabel132 = new javax.swing.JLabel();
jLabel133 = new javax.swing.JLabel();
jLabel134 = new javax.swing.JLabel();
jTextField46 = new javax.swing.JTextField();
jTextField49 = new javax.swing.JTextField();
jTextField53 = new javax.swing.JTextField();
jLabel135 = new javax.swing.JLabel();
jLabel136 = new javax.swing.JLabel();
jLabel137 = new javax.swing.JLabel();
jLabel138 = new javax.swing.JLabel();
jLabel139 = new javax.swing.JLabel();
jTextField54 = new javax.swing.JTextField();
jLabel140 = new javax.swing.JLabel();
jLabel141 = new javax.swing.JLabel();
jLabel142 = new javax.swing.JLabel();
jLabel144 = new javax.swing.JLabel();
jLabel145 = new javax.swing.JLabel();
jTextField55 = new javax.swing.JTextField();
jLabel146 = new javax.swing.JLabel();
jTextField56 = new javax.swing.JTextField();
jLabel148 = new javax.swing.JLabel();
btnNV = new javax.swing.JButton();
jTextField57 = new javax.swing.JTextField();
jLabel149 = new javax.swing.JLabel();
jLabel143 = new javax.swing.JLabel();
jLabel151 = new javax.swing.JLabel();
jLabel152 = new javax.swing.JLabel();
jLabel157 = new javax.swing.JLabel();
jLabel153 = new javax.swing.JLabel();
jTextField58 = new javax.swing.JTextField();
jTextField59 = new javax.swing.JTextField();
jLabel154 = new javax.swing.JLabel();
jTextField60 = new javax.swing.JTextField();
jLabel160 = new javax.swing.JLabel();
jLabel164 = new javax.swing.JLabel();
jTextField61 = new javax.swing.JTextField();
jPanel7 = new javax.swing.JPanel();
btnSimulacion = new javax.swing.JButton();
jLabel147 = new javax.swing.JLabel();
jLabel150 = new javax.swing.JLabel();
jLabel158 = new javax.swing.JLabel();
jPanel8 = new javax.swing.JPanel();
jLabel112 = new javax.swing.JLabel();
jLabel113 = new javax.swing.JLabel();
jLabel114 = new javax.swing.JLabel();
jLabel115 = new javax.swing.JLabel();
jLabel116 = new javax.swing.JLabel();
jLabel117 = new javax.swing.JLabel();
jLabel118 = new javax.swing.JLabel();
jLabel119 = new javax.swing.JLabel();
jLabel120 = new javax.swing.JLabel();
jLabel121 = new javax.swing.JLabel();
jLabel122 = new javax.swing.JLabel();
jLabel123 = new javax.swing.JLabel();
jLabel124 = new javax.swing.JLabel();
jLabel125 = new javax.swing.JLabel();
jLabel126 = new javax.swing.JLabel();
jLabel128 = new javax.swing.JLabel();
jLabel129 = new javax.swing.JLabel();
jLabel130 = new javax.swing.JLabel();
jLabel131 = new javax.swing.JLabel();
jLabel159 = new javax.swing.JLabel();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
```

```
setTitle("Conveyor Optimizer");
setResizable(false);

jTabbedPane1.setMaximumSize(new java.awt.Dimension(1010, 592));
jTabbedPane1.setMinimumSize(new java.awt.Dimension(1010, 592));

jPanel1.setMaximumSize(new java.awt.Dimension(964, 564));
jPanel1.setMinimumSize(new java.awt.Dimension(964, 564));

jLabel1.setFont(new java.awt.Font("Lucida Calligraphy", 1, 36)); // NOI18N
jLabel1.setForeground(new java.awt.Color(51, 51, 0));
jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel1.setText("CONVEYOR OPTIMIZER");

jLabel2.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel2.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/conveyoroptimizer/Imagenes/Conveyor.jpg"))); // NOI18N

btnCatala.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
btnCatala.setText("Català");
btnCatala.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnCatalaActionPerformed(evt);
    }
});

btnEspañol.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
btnEspañol.setText("Español");
btnEspañol.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnEspañolActionPerformed(evt);
    }
});

btnEnglish.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
btnEnglish.setText("English");
btnEnglish.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnEnglishActionPerformed(evt);
    }
});

btnDeutsch.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
btnDeutsch.setText("Deutsch");
btnDeutsch.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnDeutschActionPerformed(evt);
    }
});

jLabel3.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/conveyoroptimizer/Imagenes/Catala.jpg"))); // NOI18N

jLabel4.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/conveyoroptimizer/Imagenes/Español.jpg"))); // NOI18N

jLabel5.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/conveyoroptimizer/Imagenes/English.jpg"))); // NOI18N
jLabel5.setMaximumSize(new java.awt.Dimension(202, 239));
jLabel5.setMinimumSize(new java.awt.Dimension(202, 239));
jLabel5.setPreferredSize(new java.awt.Dimension(202, 239));

jLabel6.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/conveyoroptimizer/Imagenes/Deutsch.jpg"))); // NOI18N
```




```

        .addComponent(jLabel167)
        .addGap(18, 18, 18)
        .addComponent(jLabel168,           javax.swing.GroupLayout.PREFERRED_SIZE,      122,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGroup(jPanel2Layout.createSequentialGroup()
        .addComponent(jLabel166)
        .addGap(18, 18, 18)
        .addComponent(jLabel169,           javax.swing.GroupLayout.PREFERRED_SIZE,      122,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGap(0, 0, Short.MAX_VALUE))
    );
    jPanel2Layout.setVerticalGroup(
        jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel2Layout.createSequentialGroup()
            .addGap(66, 66, 66)
            .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel2Layout.createSequentialGroup()
                .addComponent(jLabel166,           javax.swing.GroupLayout.PREFERRED_SIZE,      23,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel169, javax.swing.GroupLayout.DEFAULT_SIZE, 29, Short.MAX_VALUE))
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(jLabel168,           javax.swing.GroupLayout.PREFERRED_SIZE,      23,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel167, javax.swing.GroupLayout.PREFERRED_SIZE, 0, Short.MAX_VALUE))
                .addGap(18, 18, 18)
                .addComponent(jLabel165)
                .addGap(62, 62, 62)
                .addComponent(jLabel162,           javax.swing.GroupLayout.PREFERRED_SIZE,      23,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(18, 18, 18)
                .addComponent(jLabel165)
                .addGap(62, 62, 62)
                .addComponent(jLabel162,           javax.swing.GroupLayout.PREFERRED_SIZE,      23,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(18, 18, 18)
                .addComponent(jLabel167, javax.swing.GroupLayout.PREFERRED_SIZE, 0, Short.MAX_VALUE))
                .addGap(18, 18, 18)
                .addComponent(jLabel165)
                .addGap(62, 62, 62)
                .addComponent(jLabel162,           javax.swing.GroupLayout.PREFERRED_SIZE,      23,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(18, 18, 18)
                .addComponent(jLabel167, javax.swing.GroupLayout.PREFERRED_SIZE, 0, Short.MAX_VALUE))
            )
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(jLabel168,           javax.swing.GroupLayout.PREFERRED_SIZE,      23,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel169, javax.swing.GroupLayout.PREFERRED_SIZE, 0, Short.MAX_VALUE))
            .addGap(18, 18, 18)
            .addComponent(jLabel166)
            .addGap(62, 62, 62)
            .addComponent(jLabel162,           javax.swing.GroupLayout.PREFERRED_SIZE,      23,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)
            .addComponent(jLabel165)
            .addGap(62, 62, 62)
            .addComponent(jLabel162,           javax.swing.GroupLayout.PREFERRED_SIZE,      23,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)
            .addComponent(jLabel167, javax.swing.GroupLayout.PREFERRED_SIZE, 0, Short.MAX_VALUE))
            .addGap(18, 18, 18)
            .addComponent(jLabel165)
            .addGap(62, 62, 62)
            .addComponent(jLabel162,           javax.swing.GroupLayout.PREFERRED_SIZE,      23,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)
            .addComponent(jLabel167, javax.swing.GroupLayout.PREFERRED_SIZE, 0, Short.MAX_VALUE))
        )
        .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel7, javax.swing.GroupLayout.DEFAULT_SIZE, 54, Short.MAX_VALUE)
        .addComponent(btnLeerArchivoCinta, javax.swing.GroupLayout.DEFAULT_SIZE, 54, Short.MAX_VALUE)
        .addComponent(jTextField1, javax.swing.GroupLayout.DEFAULT_SIZE, 54, Short.MAX_VALUE)
        .addComponent(jLabel8, javax.swing.GroupLayout.DEFAULT_SIZE, 54, Short.MAX_VALUE)
        .addComponent(jTextField2, javax.swing.GroupLayout.DEFAULT_SIZE, 54, Short.MAX_VALUE)
        .addComponent(jTextField3, javax.swing.GroupLayout.DEFAULT_SIZE, 54, Short.MAX_VALUE))
    )
);
jTabbedPane1.addTab(" Esquema cinta ", jPanel2);

jLabel7.setFont(new java.awt.Font("Tahoma", 2, 14)); // NOI18N
jLabel7.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel7.setText("Archivo de tipo de cinta:");

btnLeerArchivoCinta.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
btnLeerArchivoCinta.setText("Buscar");
btnLeerArchivoCinta.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnLeerArchivoCintaActionPerformed(evt);
    }
});

jTextField1.setEditable(false);
jTextField1.setBackground(new java.awt.Color(204, 204, 204));
jTextField1.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField1.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel8.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel8.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel8.setText("Fuerza máxima por tope:");

jTextField2.setEditable(false);
jTextField2.setBackground(new java.awt.Color(204, 204, 204));
jTextField2.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField2.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField2.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jTextField3.setEditable(false);
jTextField3.setBackground(new java.awt.Color(204, 204, 204));
jTextField3.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N

```

```
jTextField3.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField3.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel9.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel9.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel9.setText("Diámetro de giro:");

jTextField4.setEditable(false);
jTextField4.setBackground(new java.awt.Color(204, 204, 204));
jTextField4.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField4.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel10.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel10.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel10.setText("Distancia de carga mínima:");

jTextField5.setEditable(false);
jTextField5.setBackground(new java.awt.Color(204, 204, 204));
jTextField5.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField5.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel11.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel11.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel11.setText("Distancia de descarga mínima:");

jTextField6.setEditable(false);
jTextField6.setBackground(new java.awt.Color(204, 204, 204));
jTextField6.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField6.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel12.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel12.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel12.setText("Paso de paleta mínimo:");

jTextField7.setEditable(false);
jTextField7.setBackground(new java.awt.Color(204, 204, 204));
jTextField7.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField7.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel13.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel13.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel13.setText("Delta tope:");

jTextField8.setEditable(false);
jTextField8.setBackground(new java.awt.Color(204, 204, 204));
jTextField8.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField8.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel14.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel14.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel14.setText("Velocidad de cinta mínima:");

jLabel15.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel15.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel15.setText("Velocidad de cinta máxima:");

jTextField9.setEditable(false);
jTextField9.setBackground(new java.awt.Color(204, 204, 204));
jTextField9.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField9.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel16.setFont(new java.awt.Font("Tahoma", 1, 18)); // NOI18N
jLabel16.setText("Parámetros intrínsecos de la cinta");
```

```
jLabel17.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel17.setText("(N)");

jLabel18.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel18.setText("(mm)");

jLabel19.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel19.setText("(mm)");

jLabel20.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel20.setText("(mm)");

jLabel21.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel21.setText("(mm)");

jLabel22.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel22.setText("(mm)");

jLabel23.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel23.setText("(m/min)");

jLabel24.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel24.setText("(m/min)");

jLabel61.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel61.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel61.setText("Distancia antes del giro de vuelta:");

jTextField25.setEditable(false);
jTextField25.setBackground(new java.awt.Color(204, 204, 204));
jTextField25.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField25.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel62.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel62.setText("(mm)");

jLabel63.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel63.setText("(mm)");

jTextField26.setEditable(false);
jTextField26.setBackground(new java.awt.Color(204, 204, 204));
jTextField26.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField26.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel64.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel64.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel64.setText("Distancia antes del giro de ida:");

jLabel65.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel65.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel65.setText("Distancia despues del giro de vuelta:");

jTextField27.setEditable(false);
jTextField27.setBackground(new java.awt.Color(204, 204, 204));
jTextField27.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField27.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel66.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel66.setText("(mm)");

jLabel67.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel67.setText("(mm)");

jTextField28.setEditable(false);
```

```
jTextField28.setBackground(new java.awt.Color(204, 204, 204));
jTextField28.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField28.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel68.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel68.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel68.setText("Distancia despues del giro de ida:");

jLabel83.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel83.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel83.setText("Constante del tope espera carga:");

jTextField35.setEditable(false);
jTextField35.setBackground(new java.awt.Color(204, 204, 204));
jTextField35.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField35.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel84.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel84.setText("(mm)");

jLabel161.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel161.setText("© Juan Manuel Álvarez Enríquez, 2013");

javax.swing.GroupLayout jPanel3Layout = new javax.swing.GroupLayout(jPanel3);
jPanel3.setLayout(jPanel3Layout);
jPanel3Layout.setHorizontalGroup(
    jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel3Layout.createSequentialGroup()
            .addGap(32, 32, 32)
            .addComponent(jLabel16, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(0, 0, Short.MAX_VALUE))
        .addGroup(jPanel3Layout.createSequentialGroup()
            .addGap(53, 53, 53)
            .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(jPanel3Layout.createSequentialGroup()
                    .addGap(18, 18, 18)
                    .addComponent(jLabel8, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
                .addGroup(jPanel3Layout.createSequentialGroup()
                    .addGap(18, 18, 18)
                    .addComponent(jLabel9, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
                .addGroup(jPanel3Layout.createSequentialGroup()
                    .addGap(18, 18, 18)
                    .addComponent(jLabel12, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)))
            .addGap(276, 276, 276))
        .addGroup(jPanel3Layout.createSequentialGroup()
            .addGap(88, 88, 88)
            .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(jPanel3Layout.createSequentialGroup()
                    .addGap(18, 18, 18)
                    .addComponent(jTextField35, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(18, 18, 18)
                    .addComponent(jLabel84, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGroup(jPanel3Layout.createSequentialGroup()
                    .addGap(18, 18, 18)
                    .addComponent(jLabel11, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(18, 18, 18)
                    .addComponent(jLabel10, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(18, 18, 18)
                    .addComponent(jLabel13, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(18, 18, 18)
                    .addComponent(jLabel83, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)))
            .addGap(522, 522, 522))
        .addGroup(jPanel3Layout.createSequentialGroup()
            .addGap(276, 276, 276)
            .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(jPanel3Layout.createSequentialGroup()
                    .addGap(18, 18, 18)
                    .addComponent(jLabel11, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(18, 18, 18)
                    .addComponent(jLabel10, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(18, 18, 18)
                    .addComponent(jLabel13, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(18, 18, 18)
                    .addComponent(jLabel83, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGroup(jPanel3Layout.createSequentialGroup()
                    .addGap(18, 18, 18)
                    .addComponent(jLabel8, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(18, 18, 18)
                    .addComponent(jLabel9, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(18, 18, 18)
                    .addComponent(jLabel12, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(18, 18, 18)
                    .addComponent(jLabel68, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGroup(jPanel3Layout.createSequentialGroup()
                    .addGap(18, 18, 18)
                    .addComponent(jLabel16, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(18, 18, 18)
                    .addComponent(jLabel84, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(18, 18, 18)
                    .addComponent(jLabel161, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)))
            .addGap(88, 88, 88))
)
```



```

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jTextField6, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jTextField7, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED))

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jLabel21)
            .addComponent(jLabel22)))
            .addGroup(jPanel3Layout.createSequentialGroup()

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
            .addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED))

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jLabel17)
            .addComponent(jLabel18)))
            .addGroup(jPanel3Layout.createSequentialGroup()
                .addComponent(jTextField5, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(jLabel20))
            .addGroup(jPanel3Layout.createSequentialGroup()
                .addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(jLabel19)))
            .addGap(58, 58, 58)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
            .addComponent(jLabel61, javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jLabel14, javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jLabel15, javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jLabel68, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
            .addComponent(jLabel65, javax.swing.GroupLayout.DEFAULT_SIZE, 288,
Short.MAX_VALUE)
            .addComponent(jLabel64, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
            .addGap(18, 18, 18)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel3Layout.createSequentialGroup()

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jTextField9, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jTextField8, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED))

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jLabel24)
            .addComponent(jLabel23)))
            .addGroup(jPanel3Layout.createSequentialGroup()

```

```

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jTextField25, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jTextField28, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jTextField27, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jTextField26, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED))

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jLabel67)
            .addComponent(jLabel63)
            .addComponent(jLabel66)
            .addComponent(jLabel62))))
            .addGroup(jPanel3Layout.createSequentialGroup()
            .addGap(44, 44, 44)
            .addComponent(jLabel7, javax.swing.GroupLayout.PREFERRED_SIZE, 210,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, 589,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addComponent(btnLeerArchivoCinta, javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE)))
            .addGap(0, 163, Short.MAX_VALUE))
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel3Layout.createSequentialGroup()
            .addGap(0, 0, Short.MAX_VALUE)
            .addComponent(jLabel161, javax.swing.GroupLayout.PREFERRED_SIZE, 221,
javax.swing.GroupLayout.PREFERRED_SIZE)))
            .addContainerGap())
);

jPanel3Layout.setVerticalGroup(
    jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel3Layout.createSequentialGroup()
        .addContainerGap()
        .addComponent(jLabel16, javax.swing.GroupLayout.PREFERRED_SIZE, 31,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(55, 55, 55)
        .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jLabel7, javax.swing.GroupLayout.PREFERRED_SIZE, 31,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, 28,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(btnLeerArchivoCinta, javax.swing.GroupLayout.PREFERRED_SIZE, 31,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(68, 68, 68)
        .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel3Layout.createSequentialGroup()
                .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addGroup(jPanel3Layout.createSequentialGroup()
                        .addComponent(jLabel8, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addGap(18, 18, 18)
                        .addComponent(jLabel9, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addGap(20, 20, 20)
                        .addComponent(jLabel12, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addGap(18, 18, 18)
                        .addComponent(jLabel13, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    )
                )
            )
        )
    )
);

```

```

        .addGap(18, 18, 18)
        .addComponent(jLabel10,      javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, 18)
        .addComponent(jLabel11,      javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGroup(jPanel3Layout.createSequentialGroup())

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField28,    javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel67))
        .addGap(18, 18, 18)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField27,    javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel66))
        .addGap(18, 18, 18)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField26,    javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel63))
        .addGap(18, 18, 18)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField25,    javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel62))
        .addGap(18, 18, 18)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField8,     javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel24))
        .addGap(18, 18, 18)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField9,     javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel23)))
        .addGap(18, 18, 18)
        .addComponent(jLabel83,         javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGroup(jPanel3Layout.createSequentialGroup()
        .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel3Layout.createSequentialGroup()
        .addComponent(jPanel3Layout.createSequentialGroup())
        .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField2,     javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel17))
        .addGap(18, 18, 18)

.addComponent(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField3,     javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel18))
        .addGap(19, 19, 19)

.addComponent(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField6,     javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel19))
        .addGap(19, 19, 19)

```

```

        .addComponent(jLabel21))
        .addGap(18, 18, 18)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField7, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel22))
        .addGap(18, 18, 18)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel19))
        .addGap(18, 18, 18)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField5, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel20)))
        .addGroup(jPanel3Layout.createSequentialGroup()
        .addComponent(jLabel68, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, 18)
        .addComponent(jLabel65, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, 18)
        .addComponent(jLabel64, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, 18)
        .addComponent(jLabel61, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, 18)
        .addComponent(jLabel14, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, 18)
        .addComponent(jLabel15, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGap(18, 18, 18)
        .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField35, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel84))))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
Short.MAX_VALUE)
        .addComponent(jLabel161, javax.swing.GroupLayout.PREFERRED_SIZE, 23,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap())
);

jTabbedPane1.addTab("    Tipo de cinta    ", jPanel3);

jLabel25.setFont(new java.awt.Font("Tahoma", 1, 18)); // NOI18N
jLabel25.setText("Introducir los parámetros operativos de la cinta");

jLabel26.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
jLabel26.setText("DISTANCIAS");

jTextField10.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField10.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField10.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel27.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel27.setText("(mm)");

```

```
jLabel28.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel28.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel28.setText("Distancia entre ejes de la cinta:");

jLabel29.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel29.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel29.setText("Coordenada posición de carga:");

jTextField11.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField11.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField11.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel30.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel30.setText("(mm)");

jLabel31.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel31.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel31.setText("Coordenada posición de descarga:");

jLabel32.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel32.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel32.setText("Coordenada sensor de carga:");

jTextField13.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField13.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField13.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jTextField12.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField12.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField12.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel33.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel33.setText("(mm");

jLabel34.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel34.setText("(mm");

jLabel35.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel35.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel35.setText("Coordenada sensor de descarga:");

jTextField14.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField14.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField14.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel36.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel36.setText("(mm");

jLabel37.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel37.setText("(mm");

jTextField15.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField15.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField15.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel38.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel38.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel38.setText("Coordenada 2o tope ida:");

jLabel39.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel39.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel39.setText("Coordenada 1r tope vuelta:");

jTextField16.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
```

```
jTextField16.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField16.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel40.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel40.setText("(mm)");

jLabel41.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
jLabel41.setText("FUNCIÓN DE PULMÓN VACÍO");

jLabel42.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel42.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel42.setText("Tiempo de ciclo:");

jTextField17.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField17.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField17.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel43.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel43.setText("(s)");

jLabel44.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel44.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel44.setText("Tiempo de carga:");

jLabel45.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel45.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel45.setText("Tiempo de descarga:");

jLabel46.setFont(new java.awt.Font("Tahoma", 2, 14)); // NOI18N
jLabel46.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel46.setText("Tiempo neto:");

jTextField18.setEditable(false);
jTextField18.setBackground(new java.awt.Color(204, 204, 204));
jTextField18.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jTextField18.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField18.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jTextField19.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField19.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField19.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jTextField20.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField20.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField20.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel47.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel47.setText("(s)");

jLabel48.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel48.setText("(s)");

jLabel49.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel49.setText("(s)");

jLabel50.setText("_____");

jLabel51.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
jLabel51.setText("OTROS");

jLabel52.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel52.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel52.setText("Fuerza ejercida por paleta:");
```

```
jLabel53.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel53.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel53.setText("Paso de paleta:");

jLabel54.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel54.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel54.setText("Espacio necesario robot de carga:");

jLabel55.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel55.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel55.setText("Espacio necesario robot de descarga:");

jTextField21.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField21.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField21.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jTextField22.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField22.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField22.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jTextField23.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField23.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField23.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jTextField24.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField24.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField24.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel56.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel56.setText("(N)");

jLabel57.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel57.setText("(mm)");

jLabel58.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel58.setText("(mm)");

jLabel59.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel59.setText("(mm)");

jLabel60.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
jLabel60.setText("TIEMPOS");

chkPulmonVacio.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
chkPulmonVacio.setText("Pulmón vacío");

btnValidarParametros.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
btnValidarParametros.setText("Validar parámetros");
btnValidarParametros.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnValidarParametrosActionPerformed(evt);
    }
});

jLabel85.setFont(new java.awt.Font("Tahoma", 2, 14)); // NOI18N
jLabel85.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel85.setText("Paletas máximas por tope:");

jTextField36.setEditable(false);
jTextField36.setBackground(new java.awt.Color(204, 204, 204));
jTextField36.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField36.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField36.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));



```



```

        .addComponent(jTextField11,           javax.swing.GroupLayout.PREFERRED_SIZE,     88,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(9, 9, 9)
        .addComponent(jLabel30))
    .addGroup(jPanel4Layout.createSequentialGroup())
        .addComponent(jTextField12,           javax.swing.GroupLayout.PREFERRED_SIZE,     88,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(9, 9, 9)
        .addComponent(jLabel33))
    .addGroup(jPanel4Layout.createSequentialGroup())
        .addComponent(jTextField13,           javax.swing.GroupLayout.PREFERRED_SIZE,     88,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(9, 9, 9)
        .addComponent(jLabel34))
    .addGroup(jPanel4Layout.createSequentialGroup())
        .addComponent(jTextField10,           javax.swing.GroupLayout.PREFERRED_SIZE,     88,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(9, 9, 9)
        .addComponent(jLabel27))
    .addGroup(jPanel4Layout.createSequentialGroup())
        .addComponent(jTextField14,           javax.swing.GroupLayout.PREFERRED_SIZE,     88,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(9, 9, 9)
        .addComponent(jLabel36))
    .addGroup(jPanel4Layout.createSequentialGroup())
        .addComponent(jTextField15,           javax.swing.GroupLayout.PREFERRED_SIZE,     88,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(9, 9, 9)
        .addComponent(jLabel37))
    .addGroup(jPanel4Layout.createSequentialGroup())
        .addComponent(jTextField16,           javax.swing.GroupLayout.PREFERRED_SIZE,     88,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(9, 9, 9)
        .addComponent(jLabel40))
    .addGroup(jPanel4Layout.createSequentialGroup())
        .addComponent(jTextField24,           javax.swing.GroupLayout.PREFERRED_SIZE,     88,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(9, 9, 9)
        .addComponent(jLabel56)))
    .addGroup(jPanel4Layout.createSequentialGroup())
        .addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jLabel51,           javax.swing.GroupLayout.PREFERRED_SIZE,     240,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                .addComponent(jLabel52,           javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jLabel55,           javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jLabel35,           javax.swing.GroupLayout.PREFERRED_SIZE,     338,
javax.swing.GroupLayout.PREFERRED_SIZE)))
            .addComponent(jLabel53,           javax.swing.GroupLayout.PREFERRED_SIZE,     338,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(0, 0, Short.MAX_VALUE)))
    .addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel4Layout.createSequentialGroup()
            .addGap(116, 116, 116)
            .addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel44,           javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jLabel45,           javax.swing.GroupLayout.PREFERRED_SIZE,     88,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGroup(jPanel4Layout.createSequentialGroup()
            .addGap(116, 116, 116)
            .addComponent(jLabel46,           javax.swing.GroupLayout.PREFERRED_SIZE,     88,
javax.swing.GroupLayout.PREFERRED_SIZE)))
    .addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jLabel47,           javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jLabel48,           javax.swing.GroupLayout.PREFERRED_SIZE,     88,
javax.swing.GroupLayout.PREFERRED_SIZE)))

```

```

        .addComponent(jLabel45)           javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jLabel42,           javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        .addGap(18, 18, 18)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel4Layout.createSequentialGroup()
            .addComponent(jTextField17, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(9, 9, 9)
            .addComponent(jLabel43))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel4Layout.createSequentialGroup()
            .addComponent(jTextField20, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(9, 9, 9)
            .addComponent(jLabel47)))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel4Layout.createSequentialGroup()
            .addComponent(jTextField19, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(9, 9, 9)
            .addComponent(jLabel48)))
        .addGap(121, 121, 121))
    .addGroup(jPanel4Layout.createSequentialGroup()

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel4Layout.createSequentialGroup()
            .addComponent(jLabel46, javax.swing.GroupLayout.PREFERRED_SIZE, 220,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)
            .addComponent(jTextField18, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(9, 9, 9)
            .addComponent(jLabel49, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addComponent(jLabel60, javax.swing.GroupLayout.PREFERRED_SIZE, 194,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGroup(jPanel4Layout.createSequentialGroup()
            .addComponent(chkPulmonVacio, javax.swing.GroupLayout.PREFERRED_SIZE, 174,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(btnValidarParametros, javax.swing.GroupLayout.PREFERRED_SIZE, 216,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel41, javax.swing.GroupLayout.PREFERRED_SIZE, 305,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(108, 108, 108)))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
        .addGroup(jPanel4Layout.createSequentialGroup()
            .addComponent(jLabel50, javax.swing.GroupLayout.PREFERRED_SIZE, 377,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(81, 81, 81)))
        .addGroup(jPanel4Layout.createSequentialGroup()
            .addComponent(jLabel25, javax.swing.GroupLayout.PREFERRED_SIZE, 522,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        .addGroup(jPanel4Layout.createSequentialGroup()
            .addComponent(jLabel155, javax.swing.GroupLayout.PREFERRED_SIZE, 221,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap()))
    .addContainerGap()

```



```

        .addComponent(jLabel31,           javax.swing.GroupLayout.PREFERRED_SIZE,      25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(jLabel32,           javax.swing.GroupLayout.PREFERRED_SIZE,      25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(jLabel35,           javax.swing.GroupLayout.PREFERRED_SIZE,      25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(12, 12, 12)
        .addComponent(jLabel38,           javax.swing.GroupLayout.PREFERRED_SIZE,      25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel39,           javax.swing.GroupLayout.PREFERRED_SIZE,      25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField16,         javax.swing.GroupLayout.PREFERRED_SIZE,      25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel40)
        .addGap(26, 26, 26)
        .addComponent(jLabel51,           javax.swing.GroupLayout.PREFERRED_SIZE,      25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jLabel52,           javax.swing.GroupLayout.PREFERRED_SIZE,      25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(jLabel85,           javax.swing.GroupLayout.PREFERRED_SIZE,      25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(jLabel53,           javax.swing.GroupLayout.PREFERRED_SIZE,      25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(jLabel54,           javax.swing.GroupLayout.PREFERRED_SIZE,      25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(jLabel55,           javax.swing.GroupLayout.PREFERRED_SIZE,      25,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGroup(jPanel4Layout.createSequentialGroup()
            .addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jTextField10,         javax.swing.GroupLayout.PREFERRED_SIZE,      25,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel27))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED))

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField11,         javax.swing.GroupLayout.PREFERRED_SIZE,      25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel30))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField12,         javax.swing.GroupLayout.PREFERRED_SIZE,      25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel33))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField13,         javax.swing.GroupLayout.PREFERRED_SIZE,      25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel34))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

```

```

        .addComponent(jTextField14,      javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel36))
        .addGap(12, 12, 12)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField15,      javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel37))
        .addGap(89, 89, 89)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField24,      javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel56))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField36,      javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel86))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField23,      javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel57))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField22,      javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel58))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField21,      javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel59))))))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 38, Short.MAX_VALUE)
        .addComponent(jLabel155,      javax.swing.GroupLayout.PREFERRED_SIZE, 23,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap())
);

jTabbedPane1.addTab("    Parámetros    ", jPanel4);

jLabel69.setFont(new java.awt.Font("Tahoma", 1, 18)); // NOI18N
jLabel69.setText("Resultados basados en la geometría de la cinta");

jLabel70.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
jLabel70.setText("IDA");

jLabel71.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel71.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);

jLabel72.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel72.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel72.setText("Número de topes:");

jLabel73.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel73.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel73.setText("Coordenadas de los topes:");

```

```
jLabel74.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel74.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel74.setText("Número de sensores:");

jTextField29.setEditable(false);
jTextField29.setBackground(new java.awt.Color(204, 204, 204));
jTextField29.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField29.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jTextField30.setEditable(false);
jTextField30.setBackground(new java.awt.Color(204, 204, 204));
jTextField30.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField30.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField30.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jTextField31.setEditable(false);
jTextField31.setBackground(new java.awt.Color(204, 204, 204));
jTextField31.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField31.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jTextField32.setEditable(false);
jTextField32.setBackground(new java.awt.Color(204, 204, 204));
jTextField32.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField32.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField32.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jTextField33.setEditable(false);
jTextField33.setBackground(new java.awt.Color(204, 204, 204));
jTextField33.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField33.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField33.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel96.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel96.setText("*Distancias en mm y referenciadas al eje de vuelta");

jLabel97.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel97.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel97.setText("Coordenadas de los sensores:");

jLabel88.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
jLabel88.setText("TOTAL");

jLabel89.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel89.setText("Número de paletas máximo:");

jLabel90.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel90.setText("Número de topes:");

jTextField37.setEditable(false);
jTextField37.setBackground(new java.awt.Color(204, 204, 204));
jTextField37.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField37.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField37.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jTextField38.setEditable(false);
jTextField38.setBackground(new java.awt.Color(204, 204, 204));
jTextField38.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField38.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField38.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jTextField39.setEditable(false);
jTextField39.setBackground(new java.awt.Color(204, 204, 204));
jTextField39.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField39.setHorizontalAlignment(javax.swing.JTextField.CENTER);
```

```

jTextField39.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel91.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel91.setText("Número de sensores:");

btnResultadosGeometricos.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
btnResultadosGeometricos.setText("Calcular resultados");
btnResultadosGeometricos.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnResultadosGeometricosActionPerformed(evt);
    }
});

jLabel81.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel81.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel81.setText("Número de paletas por tope:");

jTextField34.setEditable(false);
jTextField34.setBackground(new java.awt.Color(204, 204, 204));
jTextField34.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField34.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField34.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

chkRepartirTopes_ida.setText("Equirrepartir topes");

ComboBxPenultim_ida.setModel(new javax.swing.DefaultComboBoxModel(new String[] {"Última", "Penúltima"}));

jLabel87.setText("Sensor en paleta:");

jLabel95.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel95.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel95.setText("Número de paletas en el último tope:");

jTextField47.setEditable(false);
jTextField47.setBackground(new java.awt.Color(204, 204, 204));
jTextField47.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField47.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField47.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

chkRepartirTopes_v.setText("Equirrepartir topes");

jLabel82.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel82.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel82.setText("Número de paletas por tope:");

jTextField40.setEditable(false);
jTextField40.setBackground(new java.awt.Color(204, 204, 204));
jTextField40.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField40.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField40.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jTextField48.setEditable(false);
jTextField48.setBackground(new java.awt.Color(204, 204, 204));
jTextField48.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField48.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField48.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel98.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel98.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel98.setText("Número de paletas en el último tope:");

jLabel75.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel75.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);

```

```
jLabel75.setText("Número de topes:");
jLabel76.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel76.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel76.setText("Número de paletas en cola:");

jLabel77.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel77.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel77.setText("Número de sensores:");

jLabel99.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel99.setText("*Distancias en mm y referenciadas al eje de ida");

jLabel78.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel78.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel78.setText("Coordenadas de los topes:");

jTextField41.setEditable(false);
jTextField41.setBackground(new java.awt.Color(204, 204, 204));
jTextField41.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField41.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField41.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jTextField42.setEditable(false);
jTextField42.setBackground(new java.awt.Color(204, 204, 204));
jTextField42.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField42.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField42.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel79.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
jLabel79.setText("VUELTA");

ComboBxPenultim_v.setModel(new javax.swing.DefaultComboBoxModel(new String[] {"Última", "Penúltima"}));

jLabel100.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel100.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel100.setText("Coordenadas de los sensores:");

jTextField43.setEditable(false);
jTextField43.setBackground(new java.awt.Color(204, 204, 204));
jTextField43.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField43.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField43.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jTextField44.setEditable(false);
jTextField44.setBackground(new java.awt.Color(204, 204, 204));
jTextField44.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField44.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel92.setText("Sensor en paleta:");

jTextField45.setEditable(false);
jTextField45.setBackground(new java.awt.Color(204, 204, 204));
jTextField45.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField45.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel80.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel80.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel80.setText("Distancia entre ejes:");

jTextField50.setEditable(false);
jTextField50.setBackground(new java.awt.Color(204, 204, 204));
jTextField50.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
```

```
jTextField50.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField50.setBorderStyle(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel93.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel93.setText("(mm)");

jLabel94.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel94.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel94.setText("Paso de paleta:");

jTextField51.setEditable(false);
jTextField51.setBackground(new java.awt.Color(204, 204, 204));
jTextField51.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField51.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField51.setBorderStyle(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel102.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel102.setText("(mm");

jLabel101.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel101.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel101.setText("Número de paletas en cola:");

jLabel103.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel103.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel103.setText("Número de paletas en descarga:");

jLabel104.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel104.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel104.setText("Número de paletas en carga:");

jLabel105.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel105.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);

jTextField52.setEditable(false);
jTextField52.setBackground(new java.awt.Color(204, 204, 204));
jTextField52.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField52.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField52.setBorderStyle(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel106.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel106.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel106.setText("Número de paletas total:");

jLabel107.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel107.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);

jLabel108.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel108.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel108.setText("Número de paletas máximo por tope:");

jLabel109.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel109.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel109.setText("Número de paletas máximo por tope:");

jLabel110.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel110.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);

jLabel111.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel111.setText("Pulmón vacío:");

jLabel156.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel156.setText("© Juan Manuel Álvarez Enríquez, 2013");
```



```

        .addComponent(jLabel104, javax.swing.GroupLayout.Alignment.TRAILING,
        javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jLabel101, javax.swing.GroupLayout.Alignment.TRAILING,
        javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jLabel103, javax.swing.GroupLayout.Alignment.TRAILING,
        javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jLabel95, javax.swing.GroupLayout.PREFERRED_SIZE, 254,
        javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel73, javax.swing.GroupLayout.DEFAULT_SIZE,
        javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jLabel74, javax.swing.GroupLayout.DEFAULT_SIZE,
        javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jLabel97, javax.swing.GroupLayout.PREFERRED_SIZE, 254,
        javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(18, 18, 18)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jLabel71, javax.swing.GroupLayout.PREFERRED_SIZE, 91,
        javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField31, javax.swing.GroupLayout.PREFERRED_SIZE, 275,
        javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField32, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
        javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField33, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
        javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField30, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
        javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField34, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
        javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel107, javax.swing.GroupLayout.PREFERRED_SIZE, 91,
        javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField52, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
        javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel105, javax.swing.GroupLayout.PREFERRED_SIZE, 90,
        javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField47, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
        javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField29, javax.swing.GroupLayout.PREFERRED_SIZE, 275,
        javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(ComboBxPenultim_ida,
        javax.swing.GroupLayout.PREFERRED_SIZE, 125, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(chkRepartirTipos_ida,
        javax.swing.GroupLayout.PREFERRED_SIZE, 200, javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
        .addComponent(jLabel99, javax.swing.GroupLayout.PREFERRED_SIZE, 342,
        javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
        jPanel5Layout.createSequentialGroup())
        .addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
            .addComponent(jLabel77, javax.swing.GroupLayout.DEFAULT_SIZE,
            javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jLabel100, javax.swing.GroupLayout.PREFERRED_SIZE, 233,
            javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(18, 18, 18))

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jTextField43, javax.swing.GroupLayout.PREFERRED_SIZE, 88,
        javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField44, javax.swing.GroupLayout.PREFERRED_SIZE, 275,
        javax.swing.GroupLayout.PREFERRED_SIZE)))

```

```

.addGroup(javax.swing.GroupLayout.Alignment.LEADING,
jPanel5Layout.createSequentialGroup()

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
            .addComponent(jLabel78, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jLabel76, javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jLabel75, javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jLabel82, javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jLabel98, javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jLabel109, javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.PREFERRED_SIZE, 233, javax.swing.GroupLayout.PREFERRED_SIZE))

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel5Layout.createSequentialGroup()
                .addGap(18, 18, 18)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
            .addComponent(jTextField40, javax.swing.GroupLayout.PREFERRED_SIZE,
88, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jTextField41, javax.swing.GroupLayout.PREFERRED_SIZE,
88, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jTextField42, javax.swing.GroupLayout.PREFERRED_SIZE,
88, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jTextField48, javax.swing.GroupLayout.PREFERRED_SIZE,
88, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel110, javax.swing.GroupLayout.PREFERRED_SIZE, 91,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(ComboBxPenultim_v,
javax.swing.GroupLayout.PREFERRED_SIZE, 125, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(chkRepartirTopes_v,
javax.swing.GroupLayout.DEFAULT_SIZE, 195, Short.MAX_VALUE)
            .addComponent(jLabel92, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
            .addGroup(jPanel5Layout.createSequentialGroup()
                .addGap(18, 18, 18)
                .addComponent(jTextField45, javax.swing.GroupLayout.PREFERRED_SIZE, 275,
javax.swing.GroupLayout.PREFERRED_SIZE)))))

.addGroup(jPanel5Layout.createSequentialGroup()
            .addGap(25, 25, 25)
            .addComponent(jLabel70, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(371, 371, 371)
            .addComponent(jLabel79, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGroup(jPanel5Layout.createSequentialGroup()
                .addGap(25, 25, 25)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jLabel88, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGroup(jPanel5Layout.createSequentialGroup()
                .addGap(10, 10, 10)
                .addComponent(jLabel89)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(jTextField37, javax.swing.GroupLayout.PREFERRED_SIZE,
88, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(70, 70, 70)
                .addComponent(jLabel90)

```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
.addComponent(jTextField38, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addGap(69, 69, 69)
.addComponent(jLabel91)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
.addComponent(jTextField39, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
(javax.swing.GroupLayout.PREFERRED_SIZE)))
.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)))
.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel5Layout.createSequentialGroup())
.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
.addComponent(jLabel156, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
.addContainerGap() 88,
);
jPanel5Layout.setVerticalGroup(
jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(jPanel5Layout.createSequentialGroup()
.addContainerGap()
.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
.addComponent(jLabel69, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addComponent(jTextField50, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addComponent(jLabel93)
.addComponent(jLabel80, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addComponent(jLabel94, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addComponent(jTextField51, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addComponent(jLabel102))
.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(jPanel5Layout.createSequentialGroup()
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addComponent(btnResultadosGeometricos, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addComponent(jLabel111, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
.addGroup(jPanel5Layout.createSequentialGroup()
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
Short.MAX_VALUE)
.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
.addComponent(jTextField38, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addComponent(jLabel91)
.addComponent(jTextField39, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
.addGroup(jPanel5Layout.createSequentialGroup()
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
.addComponent(ComboBxPenultim_v, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addComponent(jPanel5Layout.createSequentialGroup()
.addComponent(jLabel92)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
.addComponent(ComboBxPenultim_ida, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED))
.addComponent(jTextField38, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)))
.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(jPanel5Layout.createSequentialGroup()
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
.addComponent(jLabel70, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addComponent(chkRepartirTopes_ida))
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)) 221,
);
});
```

```

        .addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jLabel104, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel105, javax.swing.GroupLayout.PREFERRED_SIZE, 18,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(3, 3, 3)
        .addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jLabel101, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jTextField33, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(4, 4, 4)
        .addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jLabel103, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel107, javax.swing.GroupLayout.PREFERRED_SIZE, 18,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel5Layout.createSequentialGroup()
                .addComponent(jLabel106, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(jLabel72, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(jLabel81, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGroup(jPanel5Layout.createSequentialGroup()
                .addComponent(jLabel52, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(jLabel32, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(jLabel34, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel47, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel95, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(4, 4, 4)
            .addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel108, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel71, javax.swing.GroupLayout.PREFERRED_SIZE, 18,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel73, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jTextField31, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addGroup(jPanel5Layout.createSequentialGroup()
            .addComponent(jLabel74, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED))

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jLabel73, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jTextField31, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(jLabel74, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

```

```

        .addComponent(jLabel97,           javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField29,         javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGroup(jPanel5Layout.createSequentialGroup()
        .addGap(36, 36, 36)
        .addComponent(jTextField30,         javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGroup(jPanel5Layout.createSequentialGroup())
        .addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel79,           javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(chkRepartirTopes_v))
        .addGap(18, 18, 18)
        .addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jLabel76,           javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(jLabel75,           javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGroup(jPanel5Layout.createSequentialGroup()
        .addComponent(jTextField42,         javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(jTextField41,         javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jLabel82,           javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField40,         javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(11, 11, 11)
        .addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel98,           javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField48,         javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jLabel110,          javax.swing.GroupLayout.PREFERRED_SIZE, 18,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel109,          javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel78,           javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField45,         javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jLabel77,           javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGroup(jPanel5Layout.createSequentialGroup()
        .addComponent(jLabel100,          javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGroup(jPanel5Layout.createSequentialGroup()
        .addComponent(jTextField43,         javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

```

```

        .addComponent(jTextField44, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jLabel99, javax.swing.GroupLayout.PREFERRED_SIZE, 18,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jLabel96, javax.swing.GroupLayout.PREFERRED_SIZE, 18,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, 18)
        .addComponent(jLabel88, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jTextField37, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel91, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jTextField39, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel90, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jTextField38, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel89, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addComponent(jLabel156, javax.swing.GroupLayout.PREFERRED_SIZE, 23,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap())
    );
}

jTabbedPane1.addTab(" Resultados geométricos ", jPanel5);

btnVN.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
btnVN.setText("Calcular");
btnVN.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnVNACTIONPERFORMED(evt);
    }
});
});

jLabel127.setFont(new java.awt.Font("Tahoma", 1, 18)); // NOI18N
jLabel127.setText("Número de paletas sugerido");

jLabel132.setFont(new java.awt.Font("Tahoma", 2, 14)); // NOI18N
jLabel132.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel132.setText("Velocidad mínima:");

jLabel133.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel133.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel133.setText("Velocidad deseada:");

jLabel134.setFont(new java.awt.Font("Tahoma", 2, 14)); // NOI18N
jLabel134.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel134.setText("Velocidad máxima:");

jTextField46.setEditable(false);
jTextField46.setBackground(new java.awt.Color(204, 204, 204));
jTextField46.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField46.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField46.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

```

```
jTextField49.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField49.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField49.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jTextField53.setEditable(false);
jTextField53.setBackground(new java.awt.Color(204, 204, 204));
jTextField53.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField53.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField53.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel135.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel135.setText("(m/min)");

jLabel136.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel136.setText("(m/min");

jLabel137.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel137.setText("(m/min");

jLabel138.setText("_____");
jLabel139.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
jLabel139.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel139.setText("Número de paletas sugerido:");

jTextField54.setEditable(false);
jTextField54.setBackground(new java.awt.Color(204, 204, 204));
jTextField54.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField54.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField54.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel140.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel140.setText("(paletas)");

jLabel141.setFont(new java.awt.Font("Tahoma", 1, 18)); // NOI18N
jLabel141.setText("Velocidad sugerida");

jLabel142.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel142.setText("(paletas");

jLabel144.setText("_____");

jLabel145.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
jLabel145.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel145.setText("Velocidad sugerida:");

jTextField55.setEditable(false);
jTextField55.setBackground(new java.awt.Color(204, 204, 204));
jTextField55.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField55.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField55.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel146.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel146.setText("(m/min");

jTextField56.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField56.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField56.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel148.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
jLabel148.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel148.setText("Número de paletas deseado");
```

```
btnNV.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
btnNV.setText("Calcular");
btnNV.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnNVActionPerformed(evt);
    }
});

jTextField57.setEditable(false);
jTextField57.setBackground(new java.awt.Color(204, 204, 204));
jTextField57.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField57.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField57.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel149.setFont(new java.awt.Font("Tahoma", 2, 14)); // NOI18N
jLabel149.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
jLabel149.setText("Número de paletas máximo:");

jLabel143.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel143.setText("(paletas)");

jLabel151.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel151.setText("* Pueden existir otras soluciones válidas o óptimas diferentes a la sugerida.");

jLabel152.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel152.setText("* Conveyor Optimizer proporciona una solución heurística y aproximada siempre
subestimando la capacidad del transportador.");

jLabel157.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel157.setText("© Juan Manuel Álvarez Enríquez, 2013");

jLabel153.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel153.setText("(rpm)");

jTextField58.setEditable(false);
jTextField58.setBackground(new java.awt.Color(204, 204, 204));
jTextField58.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField58.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField58.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jTextField59.setEditable(false);
jTextField59.setBackground(new java.awt.Color(204, 204, 204));
jTextField59.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField59.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField59.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel154.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel154.setText("(rpm)");

jTextField60.setEditable(false);
jTextField60.setBackground(new java.awt.Color(204, 204, 204));
jTextField60.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jTextField60.setHorizontalAlignment(javax.swing.JTextField.CENTER);
jTextField60.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel160.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel160.setText("(rpm)");

jLabel164.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel164.setText("(rpm)");

jTextField61.setEditable(false);
jTextField61.setBackground(new java.awt.Color(204, 204, 204));
```



```

.addComponent(jTextField54, javax.swing.GroupLayout.PREFERRED_SIZE, 75,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
.addComponent(jLabel140)
.addGap(16, 16, 16)))
.addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(jPanel6Layout.createSequentialGroup()
.addComponent(jTextField58, javax.swing.GroupLayout.PREFERRED_SIZE, 75,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
.addComponent(jLabel153))
.addGroup(jPanel6Layout.createSequentialGroup()

.addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
.addGroup(jPanel6Layout.createSequentialGroup()
.addComponent(jTextField60, javax.swing.GroupLayout.PREFERRED_SIZE, 75,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
.addComponent(jLabel160))
.addGroup(jPanel6Layout.createSequentialGroup()
.addComponent(jTextField59, javax.swing.GroupLayout.PREFERRED_SIZE, 75,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
.addComponent(jLabel154)))
.addGap(27, 27, 27)

.addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addComponent(jLabel149, javax.swing.GroupLayout.Alignment.TRAILING, 230, javax.swing.GroupLayout.PREFERRED_SIZE)
.addComponent(jLabel148, javax.swing.GroupLayout.Alignment.TRAILING, 230, javax.swing.GroupLayout.PREFERRED_SIZE))
.addComponent(jLabel145, javax.swing.GroupLayout.PREFERRED_SIZE, 299,
javax.swing.GroupLayout.PREFERRED_SIZE))
.addGap(18, 18, 18)

.addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(jPanel6Layout.createSequentialGroup()
.addComponent(jTextField56, javax.swing.GroupLayout.PREFERRED_SIZE, 75,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
.addComponent(jLabel142, javax.swing.GroupLayout.PREFERRED_SIZE, 75,
javax.swing.GroupLayout.PREFERRED_SIZE))
.addGroup(jPanel6Layout.createSequentialGroup()
.addComponent(jTextField57, javax.swing.GroupLayout.PREFERRED_SIZE, 75,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
.addComponent(jLabel143, javax.swing.GroupLayout.PREFERRED_SIZE, 75,
javax.swing.GroupLayout.PREFERRED_SIZE))
.addGroup(jPanel6Layout.createSequentialGroup()
.addComponent(jTextField55, javax.swing.GroupLayout.PREFERRED_SIZE, 75,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
.addComponent(jLabel146, javax.swing.GroupLayout.PREFERRED_SIZE, 60,
javax.swing.GroupLayout.PREFERRED_SIZE))
.addComponent(jTextField61, javax.swing.GroupLayout.PREFERRED_SIZE, 75,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
.addComponent(jLabel164))))
.addGroup(jPanel6Layout.createSequentialGroup()
.addComponent(jTextField59, javax.swing.GroupLayout.PREFERRED_SIZE, 75,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
.addComponent(jLabel154)))
.addGap(39, 39, 39)

```

```

        .addComponent(jLabel127,           javax.swing.GroupLayout.PREFERRED_SIZE, 351,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(181, 181, 181)
        .addComponent(jLabel141,           javax.swing.GroupLayout.PREFERRED_SIZE, 460,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGroup(jPanel6Layout.createSequentialGroup()
        .addGap(32, 32, 32)
        .addComponent(jLabel138,           javax.swing.GroupLayout.PREFERRED_SIZE, 526,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(50, 50, 50)
        .addComponent(jLabel144,           javax.swing.GroupLayout.PREFERRED_SIZE, 491,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addContainerGap())
        .addGroup(jPanel6Layout.createSequentialGroup()
        .addGap(315, 315, 315)
        .addComponent(btnVN,           javax.swing.GroupLayout.PREFERRED_SIZE, 113,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(btnNV,           javax.swing.GroupLayout.PREFERRED_SIZE, 113,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(185, 185, 185))
    );
    jPanel6Layout.setVerticalGroup(
        jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel6Layout.createSequentialGroup()
        .addGap(106, 106, 106)
        .addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel127,           javax.swing.GroupLayout.PREFERRED_SIZE, 31,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel141,           javax.swing.GroupLayout.PREFERRED_SIZE, 31,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(32, 32, 32)
        .addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel6Layout.createSequentialGroup()
        .addComponent(jLabel134,           javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField46,           javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel137)
        .addComponent(jTextField58,           javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel153)))
        .addGap(48, 48, 48)
        .addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel132,           javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField53,           javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel135)
        .addComponent(jTextField60,           javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel160))
        .addGap(4, 4, 4)
        .addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel138)
        .addComponent(jLabel144,           javax.swing.GroupLayout.PREFERRED_SIZE, 14,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(6, 6, 6)
        .addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField54,           javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel154))
    );

```

```

        .addComponent(jLabel140)
        .addComponent(jTextField55,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel145,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel146)
        .addComponent(jTextField61,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel164)
        .addComponent(jLabel139,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGroup(jPanel6Layout.createSequentialGroup()
        .addGap(36, 36, 36)
        .addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel6Layout.createSequentialGroup()
        .addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel6Layout.createSequentialGroup()
        .addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel6Layout.createSequentialGroup()
        .addComponent(jLabel149,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, 18)
        .addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel6Layout.createSequentialGroup()
        .addComponent(jLabel142,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel148,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jLabel154)
        .addComponent(jLabel49,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel136)
        .addComponent(jLabel133,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGap(18, 18, 18)
        .addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(btnVN,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(btnNV,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, 18)
        .addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jLabel152)
        .addComponent(jLabel151)
        .addComponent(jLabel157,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, 18)
        .addComponent(containerGap())))
        .addGap(18, 18, 18)
        .addComponent(tabbedPane1.addTab(" Resultados dinámicos ", jPanel6);
        .setBackground(new java.awt.Color(255, 255, 255));
        btnSimulacion.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
        btnSimulacion.setText("Visualizar transportador");
    );
}

```



```
jLabel113.setFont(new java.awt.Font("Tahoma", 1, 15)); // NOI18N
jLabel113.setText("Autor:");

jLabel114.setFont(new java.awt.Font("Tahoma", 0, 15)); // NOI18N
jLabel114.setText("Juan Manuel Álvarez Enríquez");

jLabel115.setFont(new java.awt.Font("Tahoma", 1, 15)); // NOI18N
jLabel115.setText("Titulación:");

jLabel116.setFont(new java.awt.Font("Tahoma", 0, 15)); // NOI18N
jLabel116.setText("Ingeniero Industrial");

jLabel117.setFont(new java.awt.Font("Tahoma", 1, 15)); // NOI18N
jLabel117.setText("Email:");

jLabel118.setFont(new java.awt.Font("Tahoma", 0, 15)); // NOI18N
jLabel118.setText("jma.90@hotmail.com");

jLabel119.setFont(new java.awt.Font("Tahoma", 1, 15)); // NOI18N
jLabel119.setText("Universidad:");

jLabel120.setFont(new java.awt.Font("Tahoma", 0, 15)); // NOI18N
jLabel120.setText("Escola Tècnica Superior d'Enginyeria Industrial de Barcelona (ETSEIB) - UPC");

jLabel121.setFont(new java.awt.Font("Tahoma", 0, 15)); // NOI18N
jLabel121.setText("Xavier Gavaldà Aràñ");

jLabel122.setFont(new java.awt.Font("Tahoma", 1, 15)); // NOI18N
jLabel122.setText("Tutor:");

jLabel123.setFont(new java.awt.Font("Tahoma", 0, 15)); // NOI18N
jLabel123.setText("España");

jLabel124.setFont(new java.awt.Font("Tahoma", 1, 15)); // NOI18N
jLabel124.setText("País:");

jLabel125.setFont(new java.awt.Font("Tahoma", 1, 15)); // NOI18N
jLabel125.setText("Año:");

jLabel126.setFont(new java.awt.Font("Tahoma", 0, 15)); // NOI18N
jLabel126.setText("2013");

jLabel128.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel128.setText("* Conveyor Optimezer ha sido desarrollado como Proyecto Final de Carrera de la titulación de Ingeniería Industrial bajo las especificaciones de la empresa alemana F.EE GmbH Automation.");

jLabel129.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel129.setText("* Esta aplicación debe ser utilizada como herramienta complementaria y de ayuda en el diseño y análisis de transportadores.");

jLabel130.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel130.setText("* El autor y el tutor del Proyecto no se responsabilizan de las posibles consecuencias que se deriven de su uso.");

jLabel131.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel131.setText("* Su uso indebido o la malinterpretación de sus resultados puede conllevar consecuencias desconocidas.");

jLabel159.setFont(new java.awt.Font("Tahoma", 2, 12)); // NOI18N
jLabel159.setText("© Juan Manuel Álvarez Enríquez, 2013");

javax.swing.GroupLayout jPanel8Layout = new javax.swing.GroupLayout(jPanel8);
jPanel8.setLayout(jPanel8Layout);
```



```

        .addComponent(jLabel131, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(23, javax.swing.GroupLayout.PREFERRED_SIZE))
    );
jPanel8Layout.setVerticalGroup(
    jPanel8Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel8Layout.createSequentialGroup()
        .addGap(39, 39, 39)
        .addComponent(jLabel112, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, 18)
        .addGroup(jPanel8Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel8Layout.createSequentialGroup()
                .addComponent(jLabel113, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(jLabel115, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            )
            .addComponent(jLabel117, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel119, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel122, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel124, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        )
        .addGroup(jPanel8Layout.createSequentialGroup()
            .addComponent(jLabel114, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addComponent(jLabel116, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel118, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel120, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel121, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel123, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        )
        .addGroup(jPanel8Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jLabel125, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel126, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        )
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, Short.MAX_VALUE)
        .addComponent(jLabel128, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jLabel129, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jLabel130, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
    )
);

```

```

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jLabel131, javax.swing.GroupLayout.PREFERRED_SIZE, 18,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(33, 33, 33)
        .addComponent(jLabel159, javax.swing.GroupLayout.PREFERRED_SIZE, 23,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap())
    );

jTabbedPane1.addTab("      Acerca de      ", jPanel8);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .addComponent(jTabbedPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 1159,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(0, 0, Short.MAX_VALUE))
    );
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jTabbedPane1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
);
pack();
}// </editor-fold>

private void btnLeerArchivoCintaActionPerformed(java.awt.event.ActionEvent evt) {

/*Actualizar el boolean archivo_leido a false pq aun no se ha leido correctamente el archivo*/
archivo_leido = false;
/*
 * Cada vez que se cambie de archivo, actualizar el boolean
 * error_validar a false pq se tienen que volver a validar correctamente
 * los parametros
 */
error_validar = true;
/*
 * Cada vez que se cambie de archivo, actualizar el boolean ResGeom a
 * false pq se tienen que volver a obtener los resultados geométricos
 */
ResGeom = false;
/*Borrar los jTextField para evitar confusiones*/
jTextField2.setText("");
jTextField3.setText("");
jTextField6.setText("");
jTextField7.setText("");
jTextField4.setText("");
jTextField5.setText("");
jTextField35.setText("");
jTextField28.setText("");
jTextField27.setText("");
jTextField26.setText("");
jTextField25.setText("");
jTextField8.setText("");
jTextField53.setText(""); // vmin de la pestaña Resultados Dinámicos
jTextField60.setText(""); // rpm_min de la pestaña Resultados Dinámicos
jTextField9.setText("");
jTextField46.setText(""); // vmax de la pestaña Resultados Dinámicos
jTextField58.setText(""); // rpm_max de la pestaña Resultados Dinámicos

/* El try-catch es debido al uso del "this.wait()" del

```

```

* método "UsarArchivo.java" usado aquí*
try {
    /* Abrimos el buscador de archivos y asignamos a path la ruta y
     * nombre del archivo*/
    javax.swing.JFileChooser j = new javax.swing.JFileChooser();
    j.showOpenDialog(j);
    String path = j.getSelectedFile().getAbsolutePath();

    /*Mostramos la ruta del archivo en su jTextField*/
    jTextField1.setText(path);

    /* Leemos valores del archivo, los guardamos en las variables y
     * los mostramos en los jTextField's */
    UsarArchivo a = new UsarArchivo();

    K1 = a.LeerValor("Fuerza maxima por tope (N):", path, idioma);
    jTextField2.setText(String.valueOf(K1));

    D = a.LeerValor("Diametro de giro (mm):", path, idioma);
    jTextField3.setText(String.valueOf(D));

    dCar_min = a.LeerValor("Distancia de carga minima (mm):", path, idioma);
    jTextField4.setText(String.valueOf(dCar_min));

    dDesc_min = a.LeerValor("Distancia de descarga minima (mm):", path, idioma);
    jTextField5.setText(String.valueOf(dDesc_min));

    pas_min = a.LeerValor("Paso de paleta minimo (mm):", path, idioma);
    jTextField6.setText(String.valueOf(pas_min));

    delt atop = a.LeerValor("Delta tope (mm):", path, idioma);
    jTextField7.setText(String.valueOf(delt atop));

    v_min = a.LeerValor("Velocidad cinta minima (m/min).:", path, idioma);
    jTextField8.setText(String.valueOf(v_min));
    jTextField53.setText(String.valueOf(v_min));
    jTextField60.setText(String.valueOf((float) (v_min * (1000.0 / (Math.PI * D)))));

    v_max = a.LeerValor("Velocidad cinta maxima (m/min):", path, idioma);
    jTextField9.setText(String.valueOf(v_max));
    jTextField46.setText(String.valueOf(v_max));
    jTextField58.setText(String.valueOf((float) (v_max * (1000.0 / (Math.PI * D)))));

    dG_ida = a.LeerValor("Distancia despues del giro de ida (mm):", path, idioma);
    jTextField28.setText(String.valueOf(dG_ida));

    dG_v = a.LeerValor("Distancia despues del giro de vuelta (mm):", path, idioma);
    jTextField27.setText(String.valueOf(dG_v));

    dEG_ida = a.LeerValor("Distancia antes del giro de ida (mm):", path, idioma);
    jTextField26.setText(String.valueOf(dEG_ida));

    dEG_v = a.LeerValor("Distancia antes del giro de vuelta (mm):", path, idioma);
    jTextField25.setText(String.valueOf(dEG_v));

    K5 = a.LeerValor("Constante del tope de espera a la carga (mm):", path, idioma);
    jTextField35.setText(String.valueOf(K5));

    /* Mensaje emergente que informa de que todos los valores
     * han sido leídos correctamente*/
    JOptionPane.showMessageDialog(null, mensaje1);
    /*Actualizar a cierto el boolean de archivo_leido*/
    archivo_leido = true;
} catch (InterruptedException ex) {

```

```

        Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
    }
}

private void btnValidarParametrosActionPerformed(java.awt.event.ActionEvent evt) {

    /*Actualizar el boolean error_validar a false pq aun no se han validado correctamente los parametros*/
    error_validar = true;
    /*Actualizar el boolean ResGeom a false pq aun no se han dado los resultados geometricos*/
    ResGeom = false;
    /*Borrar los jTextFields de pal max por tope y tNet calculados para evitar confusiones*/
    jTextField36.setText(String.valueOf(""));
    jTextField18.setText(String.valueOf(""));

    /*Comprobar si primero se ha seleccionado un archivo en Tipo de cinta*/
    if (!archivo_leido) {
        /*Si no se ha seleccionado, se muestra un mensaje y se detiene el programa*/
        JOptionPane.showMessageDialog(null, mensaje2);
        try {
            this.wait();
        } catch (InterruptedException ex) {
            Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
        }
    } else {
    }

    /*Leer los valores de los jTextFields y los guardamos en sus variables*/
    L = Double.parseDouble(jTextField10.getText());
    dCar = L - Double.parseDouble(jTextField11.getText()) - dG_ida;
    dDesc = Double.parseDouble(jTextField12.getText()) - dEG_ida;
    dSCar = L - Double.parseDouble(jTextField13.getText()) - dG_ida - dCar;
    dSDesc = L - Double.parseDouble(jTextField14.getText()) - dG_v;
    dT_ida01 = Double.parseDouble(jTextField15.getText()) - dEG_ida - dDesc;
    dT_v01 = Double.parseDouble(jTextField16.getText()) - dEG_v;
    K2 = Double.parseDouble(jTextField24.getText());
    pas = Double.parseDouble(jTextField23.getText());
    K3 = Double.parseDouble(jTextField22.getText());
    K4 = Double.parseDouble(jTextField21.getText());
    tCiclo = Double.parseDouble(jTextField17.getText());
    tCarga = Double.parseDouble(jTextField20.getText());
    tDescarga = Double.parseDouble(jTextField19.getText());

    /*VALIDAR PARAMETROS (11 RESTRICCIONES)*/
    /*Si una restriccion no se cumple se detiene la ejecución*/

    /*1. Fuerza máxima por tope debe ser mayor o igual que la fuerza ejercida por paleta*/
    if (K1 < K2) {
        JOptionPane.showMessageDialog(null, mensaje3);
        error_validar = true;
        try {
            this.wait();
        } catch (InterruptedException ex) {
            Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
        }
    } else {
    }

    /*Mostrar el numero de paletas maximo por tope en su jTextField*/
    jTextField36.setText(String.valueOf((int) (K1 / K2)));
    /*2. pas de ser >= pas_min*/
    if (pas < pas_min) {
        JOptionPane.showMessageDialog(null, mensaje4);
        error_validar = true;
        try {
            this.wait();
        }
    }
}

```

```

    } catch (InterruptedException ex) {
        Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
    }
}
} else {
}
/*3. dCar debe ser >= dCar_min*/
if (dCar < dCar_min) {
    JOptionPane.showMessageDialog(null, mensaje5);
    error_validar = true;
    try {
        this.wait();
    } catch (InterruptedException ex) {
        Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
    }
}
} else {
}
/*4. dDesc debe ser >= dDesc_min*/
if (dDesc < dDesc_min) {
    JOptionPane.showMessageDialog(null, mensaje6);
    error_validar = true;
    try {
        this.wait();
    } catch (InterruptedException ex) {
        Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
    }
}
} else {
}
/*5. dSCar debe ser >= pas/2*/
if (dSCar < (pas / 2)) {
    JOptionPane.showMessageDialog(null, mensaje7);
    error_validar = true;
    try {
        this.wait();
    } catch (InterruptedException ex) {
        Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
    }
}
} else {
}
/*6. dSDesc debe ser >= pas*/
if (dSDesc < pas / 2) {
    JOptionPane.showMessageDialog(null, mensaje8);
    error_validar = true;
    try {
        this.wait();
    } catch (InterruptedException ex) {
        Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
    }
}
} else {
}
/*7. dT_ida01 debe ser >= (paso + delta tope) y (K4+pas/2)*/
if (!(dT_ida01 >= (pas + deltatop)) & (dT_ida01 >= (K4 + pas / 2 + deltatop))) {
    JOptionPane.showMessageDialog(null, mensaje9);
    error_validar = true;
    try {
        this.wait();
    } catch (InterruptedException ex) {
        Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
    }
}
} else {
}
/*8. dT_v01 debe ser >= pas +K5*/
if (dT_v01 < pas + K5) {
    JOptionPane.showMessageDialog(null, mensaje10);
    error_validar = true;
}

```

```

try {
    this.wait();
} catch (InterruptedException ex) {
    Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
}
} else {
}
/*9. Distancia entre ejes L minima lado ida*/
if (L < (dEG_ida + dDesc + dT_ida01 + dG_ida + dCar + Math.max(dSCar, K3) + pas)) {
    JOptionPane.showMessageDialog(null, mensaje11);
    error_validar = true;
    try {
        this.wait();
    } catch (InterruptedException ex) {
        Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
    }
} else {
}
/*10. Distancia entre ejes L minima lado vuelta*/
if (L < (dG_v + dSDesc + dEG_v + dT_v01 + pas)) {
    JOptionPane.showMessageDialog(null, mensaje12);
    error_validar = true;
    try {
        this.wait();
    } catch (InterruptedException ex) {
        Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
    }
} else {
}
/*11. Tiempo de ciclo mayor a la suma del tiempo de carga y de descarga*/
if (tCiclo <= tCarga + tDescarga) {
    JOptionPane.showMessageDialog(null, mensaje13);
    error_validar = true;
    try {
        this.wait();
    } catch (InterruptedException ex) {
        Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
    }
} else {
}
/*Calcular el tiempo neto y mostrarlo en su jTextField*/
tNeto = tCiclo - tCarga - tDescarga;
jTextField18.setText(String.valueOf((float) tNeto));
/*FIN VALIDAR PARAMETROS*/

/*Si todos los parametros son validos, actualizar a false el error_validar y mostrar mensaje de confirmacion*/
error_validar = false;
JOptionPane.showMessageDialog(null, mensaje14);
}

private void btnResultadosGeometricosActionPerformed(java.awt.event.ActionEvent evt) {

    // Actualizar el boolean ResGeom a false pq aun no se han dado los resultados geometricos/
    ResGeom = false;
    // Borrar los jTextField's de L y paso depaleta para evitar conusiones/
    jTextField50.setText("");
    jTextField51.setText("");
    // Borrar los jLabels de pal en carga y descarga para evitar confusiones/
    jLabel105.setText("");
    jLabel107.setText("");
    // Borrar los jLabel's de pal max por tope para evitar confusiones/
    jLabel71.setText("");
    jLabel110.setText("");
}

```

```

// Borrar el jLabel de de pulmón vacío para evitar confusiones*/
jLabel111.setText(mensaje20);
// Borrar resultados antiguos de los jTextFields
jTextField33.setText("");
jTextField52.setText("");
jTextField32.setText("");
jTextField34.setText("");
jTextField47.setText("");
jTextField31.setText("");
jTextField30.setText("");
jTextField29.setText("");
jTextField42.setText("");
jTextField41.setText("");
jTextField40.setText("");
jTextField48.setText("");
jTextField45.setText("");
jTextField43.setText("");
jTextField44.setText("");
jTextField37.setText("");
jTextField38.setText("");
jTextField39.setText("");
jTextField57.setText(""); // de la pestaña "Resultados Dinámicos"
/*
 * Comprobar si primero se ha introducido un archivo valido en Tipo de
 * Cinta y si se han introducido y validado los parametros
 */
if (!archivo_leido) {
    // Si no se ha seleccionado, se muestra un mensaje y se detiene el programa*/
    JOptionPane.showMessageDialog(null, mensaje2);
    try {
        this.wait();
    } catch (InterruptedException ex) {
        Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
    }
} else {
}
if (error_validar) {
    // Si no se han validado correctamente, se muestra un mensaje y se detiene el programa*/
    JOptionPane.showMessageDialog(null, mensaje15);
    try {
        this.wait();
    } catch (InterruptedException ex) {
        Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
    }
} else {
}

// Mostrar L en un jTextField*/
jTextField50.setText(String.valueOf(L));
// Mostrar paso de paleta en un jTextField*/
jTextField51.setText(String.valueOf(pas));
// Mostrar pal en carga y descarga*/
jLabel105.setText("1");
jLabel107.setText("1");
// Mostrar pal max por tope en sus jLabel's*/
jLabel71.setText(String.valueOf((int) (K1 / K2)));
jLabel110.setText(String.valueOf((int) (K1 / K2)));
// Mostrar si Pulmón vacío activado o no en su jLabel*/
if (chkPulmonVacio.isSelected()) {
    jLabel111.setText(mensaje21);
} else if (!chkPulmonVacio.isSelected()) {
    jLabel111.setText(mensaje22);
}

```

```

// IDA

// NUM PALETAS MAX IDA (provisional hasta que se encuentra Ntop_ida)
Nmax_ida = (int) ((L - (dEG_ida + dDesc + dT_ida01) - (dG_ida + dCar + Math.max(dSCar, K3)) + (pas / 2)) / pas);

/*NUM TOPES IDA (y num pal ida definitivo)*/
/*Calcular num topes ida y mostrarlo en su jTextField y mostrar Nmax_ida definitivo en su jTextField*/
if ((Nmax_ida * K2) <= K1) { /*Si 1 tope solo aguanta todas las paletas*/
    Ntop_ida = 1; /*Nmax_ida provisional es el definitivo i Ntop_ida=1 definitivamente*/
    tamgrups_ida = Nmax_ida;
} else if ((Nmax_ida * K2) > K1) {
    tamgrups_ida = (int) (K1 / K2);
    int Nmax_ida_new = -1;
    int cuenta_whiles = 0; // contamos el num de iteraciones del while para NO entrar en un bucle infinito
    //una especie de histeresis
    int iteraciones_max = 3; // NO ponemos 1 para dar un cierto margen
    while ((Nmax_ida != Nmax_ida_new) & (cuenta_whiles <= iteraciones_max)) { /*iterar para encontrar
    Nmax_ida i Ntop_ida definitivos*/
        if (Nmax_ida_new == -1) {
            /*entramos al siguiente if con Nmax_ida (esta condicion solo se cumple en la primera iteracion)*/
        } else if (Nmax_ida_new > -1) {
            Nmax_ida = Nmax_ida_new; /*entramos al siguiente if con Nmax_ida_new*/
        }
        if ((Nmax_ida % tamgrups_ida) == 0) {
            Ntop_ida = Nmax_ida / tamgrups_ida; /*Ntop_ida = num grupos llenos*/
        } else if ((Nmax_ida % tamgrups_ida) != 0) {
            Ntop_ida = (Nmax_ida / tamgrups_ida) + 1; /*Ntop_ida = num grupos llenos + 1*/
        }
        Nmax_ida_new = (int) ((L - (dEG_ida + dDesc + dT_ida01) - (dG_ida + dCar + Math.max(dSCar, K3)) + (pas / 2) - ((Ntop_ida - 1) * deltatop)) / pas);
        // -1 pq el deltatop del 1r tope ya se ha tenido en cuenta en la 6a restriccion
        if (Math.abs(Nmax_ida - Nmax_ida_new) == 1) {
            cuenta_whiles = cuenta_whiles + 1;
        }
    }
    if (cuenta_whiles > iteraciones_max) {
        Nmax_ida = Math.min(Nmax_ida, Nmax_ida_new); //fijamos Nmax_ida al mínimo (pq es el que cabe al
        poner los topes)
        // recalculamos Ntop_ida con el Nmax_ida que se ha fijado
        if ((Nmax_ida % tamgrups_ida) == 0) {
            Ntop_ida = Nmax_ida / tamgrups_ida; /*Ntop_ida = num grupos llenos*/
        } else if ((Nmax_ida % tamgrups_ida) != 0) {
            Ntop_ida = (Nmax_ida / tamgrups_ida) + 1; /*Ntop_ida = num grupos llenos + 1*/
        }
    }
}
jTextField33.setText(String.valueOf(Nmax_ida));
jTextField52.setText(String.valueOf(Nmax_ida + 2)); /*Mostrar num paletas maximas ida en su jTextField*/
jTextField32.setText(String.valueOf(Ntop_ida + 2)); /*+2: tope en carga y tope en descarga*/
Narriba = Nmax_ida + 2;

/*DISTANCIAS TOPES IDA (y tamgrups_ida definitivo)*/
/*Si el CheckBox esta seleccionado, equirrepartir los topes entre las paletas*/
if (!chkRepartirTopes_ida.isSelected()) {
    /*Si el CheckBox NO esta seleccionado, no equirrepartimos*/
} else if (chkRepartirTopes_ida.isSelected()) {
    if (Ntop_ida == 1) {
        /*si solo hay 1 tope NO hay que equirrepartir topes*/
    } else if (Ntop_ida > 1) { /*equirrepartimos topes*/
        if ((Nmax_ida % Ntop_ida) == 0) {
            tamgrups_ida = Nmax_ida / Ntop_ida;
        } else if ((Nmax_ida % Ntop_ida) != 0) {
            tamgrups_ida = (Nmax_ida / Ntop_ida) + 1;
        }
    }
}

```

```

        }
    }

/*Mostrar el tamgrups_ida definitivo en su jTextField (Numero de paletas por tope)*/
jTextField34.setText(String.valueOf(tamgrups_ida));
/*Crear vector de (Ntop_ida) componentes*/
double dT_ida[] = new double[Ntop_ida];
/*Asignar al vector las distancias referenciadas al eje de vuelta a las que esta cada tope*/
dT_ida[0] = dEG_ida + dDesc + dT_ida01;
int i = 1;
while (i <= Ntop_ida - 1) {
    dT_ida[i] = dT_ida[0] + (tamgrups_ida * pas + deltatop) * i;
    i = i + 1;
}
/*Escribir las coordenadas de cada tope de ida en su jTextField*/
coordenadasTop_ida = " " + String.valueOf(dEG_ida + dDesc); /*coordenada tope de descarga*/
/*coordenadas topes de la cola:/
i = 0;
while (i <= Ntop_ida - 1) {
    coordenadasTop_ida = coordenadasTop_ida + " " + String.valueOf(dT_ida[i]);
    i = i + 1;
}
coordenadasTop_ida = coordenadasTop_ida + " " + String.valueOf(L - dG_ida - dCar); /*coordeanda tope
de carga*/
jTextField31.setText(coordenadasTop_ida);

/*NUM SENSORES IDA*/
/*Calcular el numero total de sensores ida y mostrarlo en su jTextField*/
NSens_ida = Ntop_ida + 1 + 1 + 1;
//+1 sensor de fin de carga, +1 sensor en posicion carga, + 1 sensor en posicion descarga
jTextField30.setText(String.valueOf(NSens_ida));

/*DISTANCIAS SENSORES DE TOPE IDA*/
/*Calcular distancias sensores de topes ida referenciadas al eje de vuelta y mostrarlas en su jTextField*/
/*Calcular num paletas del ultimo grupo (i.e. que aguanta el ultimo tope)*/
if (Ntop_ida == 1) {
    Nultimgrup_ida = Nmax_ida;
} else if (Ntop_ida > 1) {
    if ((Nmax_ida % tamgrups_ida) == 0) {
        Nultimgrup_ida = tamgrups_ida; /*num paletas ultimo grupo si esta lleno*/
    } else if ((Nmax_ida % tamgrups_ida) != 0) {
        Nultimgrup_ida = Nmax_ida - ((Nmax_ida / tamgrups_ida) * tamgrups_ida); /*num paletas ultimo
grupo si NO esta lleno*/
    }
}
/*Mostrar el numero de paletas en el ultimo tope en su jTextField*/
jTextField47.setText(String.valueOf(Nultimgrup_ida));
/*Actualizar penultimapal_ida a true si el sensor se pone en la penultima paleta*/
penultimapal_ida = (ComboBxPenultim_ida.getSelectedItem() == ComboBox2);
/*Crear vector de (Ntop_ida) componentes (tantos sensores de tope como topes)*/
double dSens_ida[] = new double[Ntop_ida];
/*Asignar al vector las distancias referenciadas al eje de vuelta a las que esta cada sensor de tope*/
if (penultimapal_ida == false) { /*si el sensor se situa en la ULTIMA paleta del grupo*/
    if (Ntop_ida == 1) {
        dSens_ida[0] = dT_ida[0] + pas * (Nultimgrup_ida - 1);
    } else if (Ntop_ida > 1) {
        i = 0;
        while (i <= (Ntop_ida - 2)) {
            dSens_ida[i] = dT_ida[i] + pas * (tamgrups_ida - 1);
            i = i + 1;
        }
        dSens_ida[Ntop_ida - 1] = dT_ida[Ntop_ida - 1] + pas * (Nultimgrup_ida - 1);
    }
}

```

```

} else if (penultimapal_ida == true) { /*si el sensor se siua en la PENULTIMA paleta del grupo*/
    if (Ntop_ida == 1) {
        dSens_ida[0] = dT_ida[0] + pas * Math.max(0, Nultimgrup_ida - 2);
    } else if (Ntop_ida > 1) {
        i = 0;
        while (i <= Ntop_ida - 2) {
            dSens_ida[i] = dT_ida[i] + pas * Math.max(0, tamgrups_ida - 2);
            i = i + 1;
        }
        dSens_ida[Ntop_ida - 1] = dT_ida[Ntop_ida - 1] + pas * Math.max(0, Nultimgrup_ida - 2);
    }
}
/*Escribir las coordenadas de cada sensor de ida en su jTextField*/
coordenadasSens_ida = " " + String.valueOf(dEG_ida + dDesc); /*coordenada sensor en descarga*/
/*coordenadas sensores de tope*/
i = 0;
while (i <= Ntop_ida - 1) {
    coordenadasSens_ida = coordenadasSens_ida + " " + String.valueOf(dSens_ida[i]);
    i = i + 1;
}
coordenadasSens_ida = coordenadasSens_ida + " " + String.valueOf(L - dG_ida - dCar - dSCar);
/*coordenada sensor de carga*/
coordenadasSens_ida = coordenadasSens_ida + " " + String.valueOf(L - dG_ida - dCar); /*coordenada sensor en carga*/
jTextField29.setText(coordenadasSens_ida);

/*VUELTA*/

/*NUM PALETAS MAX VUELTA (provisional hasta que se encuentra Ntop_v)*/
Nmax_v = (int) ((L - (dEG_v + dT_v01) - (dG_v + dSDesc) + (pas / 2)) / pas);

/*NUM TOPES VUELTA (y num pal vuelta definitivo)*/
/*Calcular num topes vuelta y mostrarlo en su jTextField y mostrar Nmax_v definitivo en su jTextField*/
if ((Nmax_v * K2) <= K1) { /*Si 1 tope solo aguanta todas las paletas*/
    Ntop_v = 1; /*Nmax_v provisional es el definitivo i Ntop_v=1 definitivamente*/
    tamgrups_v = Nmax_v;
} else if ((Nmax_v * K2) > K1) { /*Si hace falta mas de 1 tope para aguantar todas las paletas*/
    tamgrups_v = (int) (K1 / K2);
    int Nmax_v_new = -1;
    int cuenta_whiles = 0; // contamos el num de iteraciones del while para NO entrar en un bucle infinito
    (una especie de histeresis)
    int iteraciones_max = 3; // NO ponemos 1 para dar un cierto margen
    while ((Nmax_v_new != Nmax_v) & (cuenta_whiles <= iteraciones_max)) { /*iterar para encontrar
    Nmax_v i Ntop_v definitivos*/
        if (Nmax_v_new == -1) {
            //entramos al siguiente if con Nmax_v (esta condicion solo se cumple en la primera iteracion)
        } else if (Nmax_v_new > -1) {
            Nmax_v = Nmax_v_new; /*entramos al siguiente if con Nmax_v_new*/
        }
        if ((Nmax_v % tamgrups_v) == 0) {
            Ntop_v = Nmax_v / tamgrups_v; /*Ntop_v = num grupos llenos*/
        } else if ((Nmax_v % tamgrups_v) != 0) {
            Ntop_v = (Nmax_v / tamgrups_v) + 1; /*Ntop_v = num grupos llenos + 1*/
        }
        Nmax_v_new = (int) ((L - (dEG_v + dT_v01) - (dG_v + dSDesc) + (pas / 2) - ((Ntop_v - 1) * deltatop)) /
    pas);
        // -1 pq el deltatop del 1r tope ya se ha tenido en cuenta
        if (Math.abs(Nmax_v - Nmax_v_new) == 1) {
            cuenta_whiles = cuenta_whiles + 1;
        }
    }
    if (cuenta_whiles > iteraciones_max) {
}
}

```

```

Nmax_v = Math.min(Nmax_v, Nmax_v_new); //fijamos Nmax_v al mínimo (pq es el que cabe al poner
los topes)
    // recalculamos Ntop_v con el Nmax_v que se ha fijado
    if ((Nmax_v % tamgrups_v) == 0) {
        Ntop_v = Nmax_v / tamgrups_v; /*Ntop_v = num grupos llenos*/
    } else if ((Nmax_v % tamgrups_v) != 0) {
        Ntop_v = (Nmax_v / tamgrups_v) + 1; /*Ntop_v = num grupos llenos + 1*/
    }
}
jTextField42.setText(String.valueOf(Nmax_v));
jTextField41.setText(String.valueOf(Ntop_v));
Nabajo = Nmax_v;

/*DISTANCIAS TOPES VUELTA (y tamgrups_v definitivo)*/
/*Si el CheckBox esta seleccionado, equirrepartir los topes entre las paletas*/
if (!chkRepartirTopes_v.isSelected()) {
    /*Si el CheckBox NO esta seleccionado, no equirrepartimos*/
} else if (chkRepartirTopes_v.isSelected()) {
    if (Ntop_v == 1) {
        /*si solo hay 1 tope NO hay que equirrepartir topes*/
    } else if (Ntop_v > 1) { /*equirrepartimos topes*/
        if ((Nmax_v % Ntop_v) == 0) {
            tamgrups_v = Nmax_v / Ntop_v;
        } else if ((Nmax_v % Ntop_v) != 0) {
            tamgrups_v = (Nmax_v / Ntop_v) + 1;
        }
    }
}
/*Mostrar el tamgrups_v definitivo en su jTextField (Numero de paletas por tope)*/
jTextField40.setText(String.valueOf(tamgrups_v));
/*Crear vector de (Ntop_v) componentes*/
double dT_v[] = new double[Ntop_v];
/*Asignar al vector las distancias referenciadas al eje de ida a las que esta cada tope*/
dT_v[0] = dEG_v + dT_v01;
i = 1;
while (i <= Ntop_v - 1) {
    dT_v[i] = dT_v[0] + (tamgrups_v * pas + deltatop) * i;
    i = i + 1;
}
/*Escribir las coordenadas de cada tope de vuelta en su jTextField*/
i = 0;
coordenadasTop_v = "";
while (i <= Ntop_v - 1) {
    coordenadasTop_v = coordenadasTop_v + " " + String.valueOf(dT_v[i]);
    i = i + 1;
}
jTextField45.setText(coordenadasTop_v);

/*NUM SENSORES VUELTA*/
/*Calcular el numero total de sensores vuelta y mostrarlo en su jTextField*/
NSens_v = Ntop_v + 1; /*+1 sensor de fin de descarga*/
jTextField43.setText(String.valueOf(NSens_v));

/*DISTANCIAS SENSORES DE TOPE VUELTA*/
/*Calcular distancias sensores de topes vuelta referenciadas al eje de ida y mostrarlas en su jTextField*/
/*Calcular num paletas del ultimo grupo (i.e. que aguanta el ultimo tope)*/
if (Ntop_v == 1) {
    Nultimgrup_v = Nmax_v;
} else if (Ntop_v > 1) {
    if ((Nmax_v % tamgrups_v) == 0) {
        Nultimgrup_v = tamgrups_v; /*num paletas ultimo grupo si esta lleno*/
    } else if ((Nmax_v % tamgrups_v) != 0) {

```

```

        Nultimgrup_v = Nmax_v - ((Nmax_v / tamgrups_v) * tamgrups_v); /*num paletas ultimo grupo si NO
esta lleno*/
    }
}
/*Mostrar el numero de paletas en el ultimo tope en su jTextField*/
jTextField48.setText(String.valueOf(Nultimgrup_v));
/*Actualizar penultimapal_v a true si el sensor se pone en la penultima paleta*/
penultimapal_v = (ComboBoxPenultim_v.getSelectedItem() == ComboBox2);
/*Crear vector de (Ntop_v) componentes (tantos sensores de tope como topes)*/
double dSens_v[] = new double[Ntop_v];
/*Asignar al vector las distancias referenciadas al eje de ida a las que esta cada sensor de tope*/
if (penultimapal_v == false) { /*si el sensor se situa en la ULTIMA paleta del grupo*/
    if (Ntop_v == 1) {
        dSens_v[0] = dT_v[0] + pas * (Nultimgrup_v - 1);
    } else if (Ntop_v > 1) {
        i = 0;
        while (i <= (Ntop_v - 2)) {
            dSens_v[i] = dT_v[i] + pas * (tamgrups_v - 1);
            i = i + 1;
        }
        dSens_v[Ntop_v - 1] = dT_v[Ntop_v - 1] + pas * (Nultimgrup_v - 1);
    }
} else if (penultimapal_v == true) { /*si el sensor se siua en la PENULTIMA paleta del grupo*/
    if (Ntop_v == 1) {
        dSens_v[0] = dT_v[0] + pas * Math.max(0, Nultimgrup_v - 2);
    } else if (Ntop_v > 1) {
        i = 0;
        while (i <= Ntop_v - 2) {
            dSens_v[i] = dT_v[i] + pas * Math.max(0, tamgrups_v - 2);
            i = i + 1;
        }
        dSens_v[Ntop_v - 1] = dT_v[Ntop_v - 1] + pas * Math.max(0, Nultimgrup_v - 2);
    }
}
/*Escribir las coordenadas de cada sensor en su jTextField*/
/*coordenadas sensores de tope*/
i = 0;
coordenadastSens_v = "";
while (i <= Ntop_v - 1) {
    coordenadastSens_v = coordenadastSens_v + " " + String.valueOf(dSens_v[i]);
    i = i + 1;
}
coordenadastSens_v = coordenadastSens_v + " " + String.valueOf(L - dG_v - dSDesc); /*coordenada
sensor de descarga*/
jTextField44.setText(coordenadastSens_v);

/*TOTALES*/
// Número de paletas máximo que caben en el transportador y mostrarlo en su jTextField
Nmax_TOT = Nmax_ida + Nmax_v + 1; /*+1 = +1 paleta en carga, +1 paleta en descarga, -1 paleta ultima
posicion cola_v*/
jTextField37.setText(String.valueOf(Nmax_TOT));
/*Numero de topes totales y mostrarlo en su jTextField*/
Ntop_TOT = Ntop_ida + 2 + Ntop_v;
jTextField38.setText(String.valueOf(Ntop_TOT));
/*Numero de sensores totales y mostrarlo en su jTextField*/
NSens_TOT = NSens_ida + NSens_v;
jTextField39.setText(String.valueOf(NSens_TOT));

/*Si Pulmón vacío está seleccionado*/
if (!chkPulmonVacio.isSelected()) {
    /*si NO está seleccionado, todo se queda igual*/
} else if (chkPulmonVacio.isSelected()) {

```

```

if (Nmax_ida + 2 < Nmax_v) {
    jTextField33.setText(String.valueOf(Nmax_ida)); /*jTextField de Num pal cola_ida*/
    jTextField42.setText(String.valueOf(Nmax_v - Nmax_ida - 2)); /*jTextField de Num pal cola_v*/
    Nabajo = Nmax_v - Nmax_ida - 2;
    jTextField52.setText(String.valueOf(Nmax_ida + 2)); /*Mostrar num paletas total ida en su jTextField*/
    Narriba = Nmax_ida + 2;
} else if (Nmax_ida + 2 >= Nmax_v) {
    jTextField33.setText(String.valueOf(Nmax_v - 2)); // jTextField de Num pal cola_ida
    jTextField42.setText(String.valueOf(0)); // jTextField de Num pal cola_v
    Nabajo = 0;
    jTextField52.setText(String.valueOf(Nmax_v)); // Mostrar num paletas total ida en su jTextField
    Narriba = Nmax_v;
}
// Número de paletas máximo que caben en el transportador con Pulmón Vacío y mostrarlo en su jTextField
Nmax_TOT = Nmax_v;
jTextField37.setText(String.valueOf(Nmax_TOT));
}
jTextField57.setText(String.valueOf(Nmax_TOT)); // Mostrar num paletas máximo en jTextField de "Resultados Dinámicos"

/*Si se obtienen los Resultados Geométricos correctamente, actualizar a true ResGeom */
ResGeom = true;
}

private void btnSimulacionActionPerformed(java.awt.event.ActionEvent evt) {
    /*Comprobar si se han obtenido Resultados Geometricos correctamente*/
    if (!ResGeom) {
        /*Si no se han obtenido, se muestra un mensaje y se detiene el programa*/
        JOptionPane.showMessageDialog(null, mensaje16);
        try {
            this.wait();
        } catch (InterruptedException ex) {
            Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
        }
    } else {
    }

    /*VARIABLES LOCALES*/
    double Long = L;
    double Diam = D; /*Se usa el Radio*/
    double Radio = Diam / 2;
    double paso = pas;

    /*ESCALAR DISTANCIAS*/
    double escala = 0.7 * jPanel7.getWidth() / Long;
    Long = Long * escala;
    Radio = Radio * escala;
    paso = paso * escala;

    /*DIBUJAR*/
    /*Dibujar dibuja = new Dibujar();*/
    int x1, Yarriba, x2, Yabajo;
    /*Circunferencia izquierda*/
    x1 = (int) ((jPanel7.getWidth() - Long) / 2 - Radio);
    Yarriba = (int) (jPanel7.getHeight() / 2 - Radio);
    Dibujar.Redonda(jPanel7.getGraphics(), x1, Yarriba, (int) (2 * Radio));
    /*Circunferencia derecha*/
    x2 = (int) (x1 + Long);
    Dibujar.Redonda(jPanel7.getGraphics(), x2, Yarriba, (int) (2 * Radio));
    /*Linea arriba*/
    x1 = (int) (x1 + Radio);
    x2 = x1 + (int) Long;
    Dibujar.Linea(jPanel7.getGraphics(), x1, Yarriba, x2, Yarriba);
}

```

```

/*Linea abajo*/
Yabajo = (int) (Yarriba + (int) 2 * Radio);
Dibujar.Linea(jPanel7.getGraphics(), x1, Yabajo, x2, Yabajo);

/*PALETAS ARRIBA*/
double Xorigen = (jPanel7.getWidth() - Long) / 2;
double coordenada = 0;
int i = 0;
/*Paleta Descarga*/
if (i < Narriba) {
    coordenada = Xorigen + escala * (dEG_ida + dDesc);
    Dibujar.PaletaArriba(jPanel7.getGraphics(), coordenada, Yarriba, paso);
    i = i + 1;
}
/*Paleta Carga*/
if (i < Narriba) {
    Dibujar.PaletaArriba(jPanel7.getGraphics(), Xorigen + Long - escala * (dG_ida + dCar), Yarriba, paso);
    i = i + 1;
}
/*Paletas cola*/
if (i < Narriba) {
    coordenada = coordenada + escala * dT_ida01;
    while (i < Narriba - Nultimgrup_ida) {
        Dibujar.PaletaArriba(jPanel7.getGraphics(), coordenada, Yarriba, paso);
        i = i + 1;
        int cont = 1;
        while ((cont < tamgrups_ida) & (i < Narriba)) {
            coordenada = coordenada + paso;
            Dibujar.PaletaArriba(jPanel7.getGraphics(), coordenada, Yarriba, paso);
            i = i + 1;
            cont = cont + 1;
        }
        coordenada = coordenada + paso + escala * deltatop;
    }
    if (i < Narriba) {
        Dibujar.PaletaArriba(jPanel7.getGraphics(), coordenada, Yarriba, paso);
        i = i + 1;
    }
    int cont = 1;
    while ((cont < Nultimgrup_ida) & (i < Narriba)) {
        coordenada = coordenada + paso;
        Dibujar.PaletaArriba(jPanel7.getGraphics(), coordenada, Yarriba, paso);
        i = i + 1;
        cont = cont + 1;
    }
}
/*PALETAS ABAJO*/
i = 0;
if (i < Nabajo) {
    Xorigen = x2;
    coordenada = Xorigen - escala * (dEG_v + dT_v01);
    while (i < Nabajo - Nultimgrup_v) {
        Dibujar.PaletaAbajo(jPanel7.getGraphics(), coordenada, Yabajo, paso);
        i = i + 1;
        int cont = 1;
        while ((cont < tamgrups_v) & (i < Nabajo)) {
            coordenada = coordenada - paso;
            Dibujar.PaletaAbajo(jPanel7.getGraphics(), coordenada, Yabajo, paso);
            i = i + 1;
            cont = cont + 1;
        }
        coordenada = coordenada - paso - escala * deltatop;
    }
    if (i < Nabajo) {
}

```

```

Dibujar.PaletaAbajo(jPanel7.getGraphics(), coordenada, Yabajo, paso);
i = i + 1;

}

int cont = 1;
while ((cont < Nultimgrup_v) & (i < Nabajo)) {
    coordenada = coordenada - paso;
    Dibujar.PaletaAbajo(jPanel7.getGraphics(), coordenada, Yabajo, paso);
    i = i + 1;
    cont = cont + 1;
}
}

private void btnCatalaActionPerformed(java.awt.event.ActionEvent evt) {
    idioma = 2;
    JOptionPane.showMessageDialog(null, "L'idioma s'ha canviat a català");
    /*Mensajes emergentes*/
    mensaje1 = "Tots els valors han estat llegits correctament";
    mensaje2 = "Si us plau, primer seleccionar un arxiu vàlid en Tipus de cinta";
    mensaje3 = "La força màxima per topall ha de ser major o igual que la força exercida per paleta";
    mensaje4 = "El pas de paleta introduïda és menor que el pas mínim d'aquest tipus de cinta";
    mensaje5 = "La distància de càrrega introduïda és menor que la distància de càrrega mínima d'aquest tipus de cinta";
    mensaje6 = "La distància de descàrrega introduïda és menor que la distància de descàrrega mínima d'aquest tipus de cinta";
    mensaje7 = "El sensor de càrrega està a menys de mig pas de paleta de la posició de càrrega";
    mensaje8 = "El sensor de descàrrega està a menys de mig pas de paleta del fi del gir de tornada";
    mensaje9 = "La distància entre la posició de descàrrega i la posició d'espera a la descàrrega és inferior a la suma del pas de paleta més el delta topall o a la suma de l'espai necessari del robot de descàrrega més mig pas més el delta topall";
    mensaje10 = "La distància entre l'inici del gir d'anada i la posició d'espera a la càrrega és inferior al pas de paleta més la constant del topall";
    mensaje11 = "La distància entre eixos introduïda és massa petita a l'anada";
    mensaje12 = "La distància entre eixos introduïda és massa petita a la tornada";
    mensaje13 = "El temps de cicle ha de ser major a la suma del temps de càrrega i el de descàrrega";
    mensaje14 = "Tots els paràmetres són compatibles amb aquest tipus de cinta";
    mensaje15 = "Si us plau, primer introduir i validar paràmetres";
    mensaje16 = "Si us plau, primer obtenir els Resultats geomètrics";
    mensaje17 = "Velocitat massa petita. No s'ha obtingut cap solució.";
    mensaje18 = "La velocitat introduïda és inferior a la velocitat mínima";
    mensaje19 = "La velocitat introduïda és superior a la velocitat màxima";
    mensaje20 = "Pulmó buit:";
    mensaje21 = "Pulmó buit: Si";
    mensaje22 = "Pulmó buit: No";
    mensaje23 = "El nombre de paletes desitjat no pot ser inferior a 1";
    mensaje24 = "El nombre de paletes desitjat no pot ser superior al nombre màxim de paletes";
    mensaje25 = "S'ha imposat la velocitat mínima";
    mensaje26 = "Nombre de paletes massa petit. La velocitat excedeix la màxima";
    mensaje27 = "Hi ha més paletes de les necessàries amb la velocitat obtinguda";
    //NOMBRES PESTAÑAS
    jTabbedPane1.setTitleAt(0, " Inici      ");
    jTabbedPane1.setTitleAt(1, " Esquema cinta      ");
    jTabbedPane1.setTitleAt(2, " Tipus de cinta      ");
    jTabbedPane1.setTitleAt(3, " Paràmetres      ");
    jTabbedPane1.setTitleAt(4, " Resultats geomètrics      ");
    jTabbedPane1.setTitleAt(5, " Resultats dinàmics      ");
    jTabbedPane1.setTitleAt(6, " Simulació      ");
    jTabbedPane1.setTitleAt(7, " Sobre      ");
    //PESTAÑA "ESQUEMA CINTA"
    jLabel169.setText("Topall");
    jLabel168.setText("Sensor");
    //PESTAÑA "TIPO DE CINTA"
    jLabel16.setText("Paràmetres intrínsecs de la cinta");
}

```

```

jLabel7.setText("Arxiu del tipus de cinta:");
btnLeerArchivoCinta.setText("Cercar");
jLabel8.setText("Força màxima per topall:");
jLabel9.setText("Diàmetre de gir:");
jLabel10.setText("Distància de càrrega mínima:");
jLabel11.setText("Distància de descàrrega mínima:");
jLabel12.setText("Pas de paleta mínim:");
jLabel13.setText("Delta topall:");
jLabel14.setText("Velocitat de cinta mínima:");
jLabel15.setText("Velocitat de cinta màxima:");
jLabel61.setText("Distància abans del gir de tornada:");
jLabel64.setText("Distància abans del gir d'anada:");
jLabel65.setText("Distància després del gir de tornada:");
jLabel68.setText("Distància després del gir d'anada:");
jLabel83.setText("Constant del topall d'espera càrrega:");
//PESTAÑA "PARÁMETROS"
jLabel25.setText("Introduir els paràmetres operatius de la cinta");
jLabel26.setText("DISTÀNCIES");
jLabel28.setText("Distància entre eixos de la cinta:");
jLabel29.setText("Coordenada posició de càrrega:");
jLabel31.setText("Coordenada posició de descàrrega:");
jLabel32.setText("Coordenada sensor de càrrega:");
jLabel35.setText("Coordenada sensor de descàrrega:");
jLabel38.setText("Coordenada 2n topall anada:");
jLabel39.setText("Coordenada 1r topall tornada:");
jLabel41.setText("FUNCIÓ DE PULMÓ BUIT");
jLabel42.setText("Temps de cicle:");
jLabel44.setText("Temps de càrrega:");
jLabel45.setText("Temps de descàrrega:");
jLabel46.setText("Temps net:");
jLabel51.setText("ALTRES");
jLabel52.setText("Força exercida per paleta:");
jLabel53.setText("Pas de paleta:");
jLabel54.setText("Espai necessari robot de càrrega:");
jLabel55.setText("Espai necessari robot de descàrrega:");
jLabel60.setText("TEMPS");
chkPulmonVacio.setText("Pulmó buit");
btnValidarParametros.setText("Validar paràmetres");
jLabel85.setText("Paletes màximes per topall:");
jLabel86.setText("(paletes)");
//PESTAÑA "RESULTADOS GEOMÉTRICOS"
jLabel111.setText(mensaje20);
jLabel69.setText("Resultats basats en la geometria de la cinta");
jLabel70.setText("ANADA");
jLabel72.setText("Nombre de topalls:");
jLabel73.setText("Coordenades dels topalls:");
jLabel74.setText("Nombre de sensors:");
jLabel96.setText("*Distancies en mm i referenciades a l'eix de tornada");
jLabel97.setText("Coordenades dels sensors:");
jLabel88.setText("TOTAL");
jLabel89.setText("Nombre de paletes màxim:");
jLabel90.setText("Nombre de topalls:");
jLabel91.setText("Nombre de sensors:");
btnResultadosGeometricos.setText("Calcular resultats");
jLabel81.setText("Nombre de paletes per topall:");
chkRepartirTopes_ida.setText("Equirrepartir topalls");
ComboBox1 = "Última";
ComboBox2 = "Penúltima";
ComboBoxPenultim_ida.setModel(new javax.swing.DefaultComboBoxModel(new String[]{ComboBox1, ComboBox2}));
jLabel87.setText("Sensor en paleta:");
jLabel95.setText("Nombre de paletes en l'últim topall:");
chkRepartirTopes_v.setText("Equirrepartir topalls");
jLabel82.setText("Nombre de paletes per topall:");

```

```

jLabel98.setText("Nombre de paletes en l'últim topall:");
jLabel75.setText("Nombre de topalls:");
jLabel76.setText("Nombre de paletes en cua:");
jLabel77.setText("Nombre de sensors:");
jLabel99.setText("*Distancies en mm i referenciades a l'eix d'anada");
jLabel78.setText("Coordenades dels topalls:");
jLabel79.setText("TORNADA");
ComboBoxPenultimo_v.setModel(new javax.swing.DefaultComboBoxModel(new String[]{ComboBox1,
ComboBox2}));
jLabel100.setText("Coordenades dels sensors:");
jLabel92.setText("Sensor en paleta:");
jLabel80.setText("Distancia entre eixos:");
jLabel94.setText("Pas de paleta:");
jLabel101.setText("Nombre de paletes en cua:");
jLabel103.setText("Nombre de paletes en descàrrega:");
jLabel104.setText("Nombre de paletes en càrrega:");
jLabel106.setText("Nombre de paletes total:");
jLabel108.setText("Nombre de paletes màxim per topall:");
jLabel109.setText("Nombre de paletes màxim per topall:");
//PESTAÑA "RESULTADOS DINÁMICOS"
btnVN.setText("Calcular");
btnNV.setText("Calcular");
jLabel127.setText("Nombre de paletes sugerit");
jLabel132.setText("Velocitat mínima:");
jLabel135.setText("(m/min)");
jLabel133.setText("Velocitat desitjada:");
jLabel136.setText("(m/min)");
jLabel134.setText("Velocitat màxima:");
jLabel137.setText("(m/min)");
jLabel139.setText("Nombre de paletes sugerit");
jLabel140.setText("(paletes)");
jLabel141.setText("Velocitat sugerida");
jLabel148.setText("Nombre de paletes desitjat:");
jLabel142.setText("(paletes)");
jLabel145.setText("Velocitat sugerida:");
jLabel146.setText("(m/min)");
jLabel149.setText("Nombre de paletes màxim:");
jLabel143.setText("(paletes)");
jLabel152.setText("* Conveyor Optimizer proporciona una solució heurística i aproximada sempre
subestimat la capacitat del transportador.");
jLabel151.setText("* Poden existir altres solicions vàlides o òptimes diferents a la sugerida.");
//PESTAÑA "SIMULACIÓN"
btnSimulacion.setText("Visualitzar transportador");
jLabel147.setText("* Es representa el nombre de paletes màxim que cap en el transportador segons les
seves característiques geomètriques.");
jLabel150.setText("* Les paletes es representen a la seva posició inicial.");
//PESTAÑA "ACERCA DE"
jLabel113.setText("Autor:");
jLabel115.setText("Titulació:");
jLabel116.setText("Enginyer Industrial");
jLabel117.setText("Email:");
jLabel119.setText("Universitat:");
jLabel122.setText("Tutor:");
jLabel124.setText("País:");
jLabel123.setText("Espanya");
jLabel125.setText("Any:");
jLabel128.setText("* Conveyor Optimizer ha estat desenvolupat com a Projecte Final de Carrera de la
titulació d'Enginyeria Industrial sota les especificacions de l'empresa alemanya F.EE GmbH Automation.");
jLabel129.setText("* Aquesta aplicació deu de ser utilitzada com a eina complementaria i d'ajut en el
disseny i anàlisi de transportadors.");
jLabel130.setText("* L'autor i el tutor del Projecte no es responsabilitzen de les possibles conseqüències
que es derivin del seu ús.");
jLabel131.setText("* El seu ús indegit o la incorrecta interpretació dels seus resultats pot comportar
conseqüències desconegudes.");

```

```

}

private void btnEspañolActionPerformed(java.awt.event.ActionEvent evt) {
    idioma = 1;
    JOptionPane.showMessageDialog(null, "El idioma ha sido cambiado a español");
    /*Mensajes emergentes/
    mensaje1 = "Todos los valores han sido leídos correctamente";
    mensaje2 = "Por favor, primero seleccionar un archivo válido en Tipo de cinta";
    mensaje3 = "La fuerza máxima por tope debe ser mayor o igual que la fuerza ejercida por paleta";
    mensaje4 = "El paso de paleta introducido es menor que el paso mínimo de este tipo de cinta";
    mensaje5 = "La distancia de carga introducida es menor que la distancia de carga mínima de este tipo de
cinta";
    mensaje6 = "La distancia de descarga introducida es menor que la distancia de descarga mínima de este
tipo de cinta";
    mensaje7 = "El sensor de carga está a menos de medio paso de paleta de la posición de carga";
    mensaje8 = "El sensor de descarga está a menos de medio paso de paleta del fin del giro de vuelta";
    mensaje9 = "La distancia entre la posición de descarga y la posición de espera a la descarga es inferior a
<paso + delta tope> o a <espacio necesario robot descarga + paso/2 + delta tope>";
    mensaje10 = "La distancia entre el inicio del giro de ida y la posición de espera a la carga es inferior a <paso
de paleta + la constante del tope>";
    mensaje11 = "La distancia entre ejes introducida es demasiado pequeña en la ida";
    mensaje12 = "La distancia entre ejes introducida es demasiado pequeña en la vuelta";
    mensaje13 = "El tiempo de ciclo debe ser mayor a la suma del tiempo de carga y de descarga";
    mensaje14 = "Todos los parámetros son compatibles con este tipo de cinta";
    mensaje15 = "Por favor, primero introducir y validar los parámetros";
    mensaje16 = "Por favor, primero obtener los Resultados geométricos";
    mensaje17 = "Velocidad demasiado pequeña. No se ha obtenido una solución.";
    mensaje18 = "La velocidad introducida es inferior a la velocidad mínima";
    mensaje19 = "La velocidad introducida es superior a la velocidad máxima";
    mensaje20 = "Pulmón vacío:";
    mensaje21 = "Pulmón vacío: Si";
    mensaje22 = "Pulmón vacío: No";
    mensaje23 = "El número de paletas deseado no puede ser inferior a 1";
    mensaje24 = "El número de paletas deseado no puede ser superior al número de paletas máximo";
    mensaje25 = "Se ha impuesto velocidad mínima";
    mensaje26 = "Número de paletas demasiado pequeño. La velocidad excede la máxima.";
    mensaje27 = "Hay más paletas de las necesarias con la velocidad obtenida";
    //NOMBRES PESTAÑAS
    jTabbedPane1.setTitleAt(0, " Inicio ");
    jTabbedPane1.setTitleAt(1, " Esquema cinta ");
    jTabbedPane1.setTitleAt(2, " Tipo de cinta ");
    jTabbedPane1.setTitleAt(3, " Parámetros ");
    jTabbedPane1.setTitleAt(4, " Resultados geométricos ");
    jTabbedPane1.setTitleAt(5, " Resultados dinámicos ");
    jTabbedPane1.setTitleAt(6, " Simulación ");
    jTabbedPane1.setTitleAt(7, " Acerca de ");
    //PESTAÑA "ESQUEMA CINTA"
    jLabel169.setText("Tope");
    jLabel168.setText("Sensor");
    //PESTAÑA "TIPO DE CINTA"
    jLabel16.setText("Parámetros intrínsecos de la cinta");
    jLabel7.setText("Archivo de tipo de cinta:");
    btnLeerArchivoCinta.setText("Buscar");
    jLabel8.setText("Fuerza máxima por tope:");
    jLabel9.setText("Diámetro de giro:");
    jLabel10.setText("Distancia de carga mínima:");
    jLabel11.setText("Distancia de descarga mínima:");
    jLabel12.setText("Paso de paleta mínimo:");
    jLabel13.setText("Delta tope:");
    jLabel14.setText("Velocidad de cinta mínima:");
    jLabel15.setText("Velocidad de cinta máxima:");
    jLabel61.setText("Distancia antes del giro de vuelta:");
    jLabel64.setText("Distancia antes del giro de ida:");
}

```

```

jLabel65.setText("Distancia despues del giro de vuelta:");
jLabel68.setText("Distancia despues del giro de ida:");
jLabel83.setText("Constante del tope espera carga:");
//PESTAÑA "PARÁMETROS"
jLabel25.setText("Introducir los parámetros operativos de la cinta");
jLabel26.setText("DISTANCIAS");
jLabel28.setText("Distancia entre ejes de la cinta:");
jLabel29.setText("Coordenada posición de carga:");
jLabel31.setText("Coordenada posición de descarga:");
jLabel32.setText("Coordenada sensor de carga:");
jLabel35.setText("Coordenada sensor de descarga:");
jLabel38.setText("Coordenada 2o tope ida:");
jLabel39.setText("Coordenada 1r tope vuelta:");
jLabel41.setText("FUNCIÓN DE PULMÓN VACÍO");
jLabel42.setText("Tiempo de ciclo:");
jLabel44.setText("Tiempo de carga:");
jLabel45.setText("Tiempo de descarga:");
jLabel46.setText("Tiempo neto:");
jLabel51.setText("OTROS");
jLabel52.setText("Fuerza ejercida por paleta:");
jLabel53.setText("Paso de paleta:");
jLabel54.setText("Espacio necesario robot de carga:");
jLabel55.setText("Espacio necesario robot de descarga:");
jLabel60.setText("TIEMPOS");
chkPulmonVacío.setText("Pulmón vacío");
btnValidarParametros.setText("Validar parámetros");
jLabel85.setText("Paletas máximas por tope:");
jLabel86.setText("(paletas)");
//PESTAÑA "RESULTADOS GEOMÉTRICOS"
jLabel111.setText(mensaje20);
jLabel69.setText("Resultados basados en la geometría de la cinta");
jLabel70.setText("IDA");
jLabel72.setText("Número de topes:");
jLabel73.setText("Coordenadas de los topes:");
jLabel74.setText("Número de sensores:");
jLabel96.setText("*Distancias en mm y referenciadas al eje de vuelta");
jLabel97.setText("Coordenadas de los sensores:");
jLabel88.setText("TOTAL");
jLabel89.setText("Número de paletas máximo:");
jLabel90.setText("Número de topes:");
jLabel91.setText("Número de sensores:");
btnResultadosGeometricos.setText("Calcular resultados");
jLabel81.setText("Número de paletas por tope:");
chkRepartirTopes_ida.setText("Equirrepartir topes");
ComboBox1 = "Última";
ComboBox2 = "Penúltima";
ComboBxPenultim_ida.setModel(new javax.swing.DefaultComboBoxModel(new String[]{ComboBox1, ComboBox2}));
jLabel87.setText("Sensor en paleta:");
jLabel95.setText("Número de paletas en el último tope:");
chkRepartirTopes_v.setText("Equirrepartir topes");
jLabel82.setText("Número de paletas por tope:");
jLabel98.setText("Número de paletas en el último tope:");
jLabel75.setText("Número de topes:");
jLabel76.setText("Número de paletas en cola:");
jLabel77.setText("Número de sensores:");
jLabel99.setText("*Distancias en mm y referenciadas al eje de ida");
jLabel78.setText("Coordenadas de los topes:");
jLabel79.setText("VUELTA");
ComboBxPenultim_v.setModel(new javax.swing.DefaultComboBoxModel(new String[]{ComboBox1, ComboBox2}));
jLabel100.setText("Coordenadas de los sensores:");
jLabel92.setText("Sensor en paleta:");
jLabel80.setText("Distancia entre ejes:");

```

```

jLabel94.setText("Paso de paleta:");
jLabel101.setText("Número de paletas en cola:");
jLabel103.setText("Número de paletas en descarga:");
jLabel104.setText("Número de paletas en carga:");
jLabel106.setText("Número de paletas total:");
jLabel108.setText("Número de paletas máximo por tope:");
jLabel109.setText("Número de paletas máximo por tope:");
//PESTAÑA "RESULTADOS DINÁMICOS"
btnVN.setText("Calcular");
btnNV.setText("Calcular");
jLabel127.setText("Número de paletas sugerido");
jLabel132.setText("Velocidad mínima:");
jLabel135.setText("(m/min)");
jLabel133.setText("Velocidad deseada:");
jLabel136.setText("(m/min)");
jLabel134.setText("Velocidad máxima:");
jLabel137.setText("(m/min)");
jLabel139.setText("Número de paletas sugerido:");
jLabel140.setText("(paletas)");
jLabel141.setText("Velocidad sugerida");
jLabel148.setText("Número de paletas deseado:");
jLabel142.setText("(paletas)");
jLabel145.setText("Velocidad sugerida:");
jLabel146.setText("(m/min)");
jLabel149.setText("Número de paletas máximo:");
jLabel143.setText("(paletas)");
jLabel152.setText("* Conveyor Optimizer proporciona una solución heurística y aproximada siempre subestimando la capacidad del transportador.");
jLabel151.setText("* Pueden existir otras soluciones válidas o óptimas diferentes a la sugerida.");
//PESTAÑA "SIMULACIÓN"
btnSimulacion.setText("Visualizar transportador");
jLabel147.setText("* Se representa el número de paletas máximo que cabe en el transportador según sus características geométricas.");
jLabel150.setText("* Las paletas se representan en su posición inicial.");
//PESTAÑA "ACERCA DE"
jLabel113.setText("Autor:");
jLabel115.setText("Titulación:");
jLabel116.setText("Ingeniero Industrial");
jLabel117.setText("Email:");
jLabel119.setText("Universidad:");
jLabel122.setText("Tutor:");
jLabel124.setText("País:");
jLabel123.setText("España");
jLabel125.setText("Año:");
jLabel128.setText("* Conveyor Optimezer ha sido desarrollado como Proyecto Final de Carrera de la titulación de Ingeniería Industrial bajo las especificaciones de la empresa alemana F.EE GmbH Automation.");
jLabel129.setText("* Esta aplicación debe ser utilizada como herramienta complementaria y de ayuda en el diseño y análisis de transportadores.");
jLabel130.setText("* El autor y el tutor del Proyecto no se responsabilizan de las posibles consecuencias que se deriven de su uso.");
jLabel131.setText("* Su uso indebido o la malinterpretación de sus resultados puede conllevar consecuencias desconocidas.");
}

private void btnEnglishActionPerformed(java.awt.event.ActionEvent evt) {
    idioma = 3;
    JOptionPane.showMessageDialog(null, "English has been established");
    /*Mensajes emergentes*/
    mensaje1 = "All values have been read correctly";
    mensaje2 = "Please first select a valid file in Type of conveyor";
    mensaje3 = "The block's maximum force must be equal or greater than the force exerted by box";
    mensaje4 = "The introduced box's step is shorter than this type of conveyor's minimum step";
    mensaje5 = "Introduced loading distance is shorter than this type of conveyor's minimum loading distance";
}

```

```

mensaje6 = "Introduced unloading distance is shorter than this type of conveyor's minimum unloading
distance";
mensaje7 = "The load sensor is less than a half-box step from load position";
mensaje8 = "The unload sensor is less than a half-box step from the end of the return turn";
mensaje9 = "The distance between unload and the waiting unload position is shorter than either <box's step
+ block's delta> or <unloading robot's space + step/2 + block's delta>";
mensaje10 = "The distance between the going turn beginning and the load waiting position is shorter than
<box's step + block constant>";
mensaje11 = "The wheelbase introduced is too short in the go";
mensaje12 = "The wheelbase introduced is too short in the return";
mensaje13 = "The cycle time must be greater than the sum of the loading time and unloading time";
mensaje14 = "All parameters are compatible with this type of conveyor";
mensaje15 = "Please introduce and validate parameters first";
mensaje16 = "Please obtain geometric results first";
mensaje17 = "Speed too low. No solution was obtained.";
mensaje18 = "Introduced speed is below the minimum speed";
mensaje19 = "Introduced speed exceeds the maximum speed";
mensaje20 = "Empty store:";
mensaje21 = "Empty store: Yes";
mensaje22 = "Empty store: No";
mensaje23 = "The requested number of boxes cannot be less than 1";
mensaje24 = "The requested number of boxes cannot exceed the maximum number of boxes";
mensaje25 = "Minimum speed was set";
mensaje26 = "The number of palettes is too small. Speed exceeds the maximum.";
mensaje27 = "With the obtained speed there are more boxes than necessary";
//NOMBRES PESTAÑAS
jTabbedPane1.setTitleAt(0, " Start ");
jTabbedPane1.setTitleAt(1, " Conveyor's diagram ");
jTabbedPane1.setTitleAt(2, " Type of conveyor ");
jTabbedPane1.setTitleAt(3, " Parameters ");
jTabbedPane1.setTitleAt(4, " Geometric results ");
jTabbedPane1.setTitleAt(5, " Dynamic results ");
jTabbedPane1.setTitleAt(6, " Simulation ");
jTabbedPane1.setTitleAt(7, " About ");
//PESTAÑA "ESQUEMA CINTA"
jLabel169.setText("Block");
jLabel168.setText("Sensor");
//PESTAÑA "TIPO DE CINTA"
jLabel16.setText("Intrinsic conveyor's parameters");
jLabel7.setText("Type of conveyor file:");
btnLeerArchivoCinta.setText("Search");
jLabel8.setText("Maximum force by block:");
jLabel9.setText("Turning diameter:");
jLabel10.setText("Minimum loading distance:");
jLabel11.setText("Minimum unloading distance:");
jLabel12.setText("Minimum box's step:");
jLabel13.setText("Block's delta:");
jLabel14.setText("Minimum conveyor's speed:");
jLabel15.setText("Maximum conveyor's speed:");
jLabel61.setText("Distance before return turn:");
jLabel64.setText("Distance before going turn:");
jLabel65.setText("Distance after return turn:");
jLabel68.setText("Distance after going turn:");
jLabel83.setText("Waiting loading block's constant:");
//PESTAÑA "PARÁMETROS"
jLabel25.setText("Introduce conveyor's operating parameters");
jLabel26.setText("DISTANCES");
jLabel28.setText("Conveyor's wheelbase:");
jLabel29.setText("Loading coordinate:");
jLabel31.setText("Unloading coordinate:");
jLabel32.setText("Loading sensor coordinate:");
jLabel35.setText("Unloading sensor coordinate:");
jLabel38.setText("2nd going block coordinate:");
jLabel39.setText("1st return block coordinate:");

```

```

jLabel41.setText("EMPTY STORE FUNCTION");
jLabel42.setText("Cycle time:");
jLabel44.setText("Loading time:");
jLabel45.setText("Unloading time:");
jLabel46.setText("Net time:");
jLabel51.setText("OTHER");
jLabel52.setText("Applied force by box:");
jLabel53.setText("Box's step:");
jLabel54.setText("Loading robot space required:");
jLabel55.setText("Unloading robot space required:");
jLabel60.setText("TIMES");
chkPulmonVacio.setText("Empty store");
btnValidarParametros.setText("Validate parameters");
jLabel85.setText("Maximum boxes by block:");
jLabel86.setText("(boxes)");
//PESTAÑA "RESULTADOS GEOMÉTRICOS"
jLabel111.setText(mensaje20);
jLabel69.setText("Results based on the conveyor's geometry");
jLabel70.setText("GOING");
jLabel72.setText("Number of blocks:");
jLabel73.setText("Block's coordinates:");
jLabel74.setText("Number of sensors:");
jLabel96.setText("*Distances in mm and referenced to the return axis");
jLabel97.setText("Sensors' coordinates:");
jLabel88.setText("TOTAL");
jLabel89.setText("Maximum number of boxes:");
jLabel90.setText("Number of blocks:");
jLabel91.setText("Number of sensors:");
btnResultadosGeometricos.setText("Compute results");
jLabel81.setText("Number of boxes by block:");
chkRepartirTopes_ida.setText("Equal-share blocks");
ComboBox1 = "Last";
ComboBox2 = "Penultimate";
ComboBxPenultim_v.setModel(new javax.swing.DefaultComboBoxModel(new String[]{ComboBox1, ComboBox2}));
jLabel87.setText("Sensor in box:");
jLabel95.setText("Number of boxes on the last block:");
chkRepartirTopes_v.setText("Equal-share blocks");
jLabel82.setText("Number of boxes by block:");
jLabel98.setText("Number of boxes in the last block:");
jLabel75.setText("Number of blocks:");
jLabel76.setText("Number of queued boxes:");
jLabel77.setText("Number of sensors:");
jLabel99.setText("*Distances in mm and referenced to the going axis");
jLabel78.setText("Blocks' coordinates:");
jLabel79.setText("RETURN");
ComboBxPenultim_v.setModel(new javax.swing.DefaultComboBoxModel(new String[]{ComboBox1, ComboBox2}));
jLabel100.setText("Sensors' coordinates:");
jLabel92.setText("Sensor in box:");
jLabel80.setText("Wheelbase:");
jLabel94.setText("Box's step:");
jLabel101.setText("Number of queued boxes:");
jLabel103.setText("Number of boxes unloading:");
jLabel104.setText("Number of boxes loading:");
jLabel106.setText("Total number of boxes:");
jLabel108.setText("Maximum number of boxes by block:");
jLabel109.setText("Maximum number of boxes by block:");
//PESTAÑA "RESULTADOS DINÁMICOS"
btnVN.setText("Compute");
btnNV.setText("Compute");
jLabel127.setText("Suggested number of boxes");
jLabel132.setText("Minimum speed:");
jLabel135.setText("(m/min)");

```

```

jLabel133.setText("Requested speed:");
jLabel136.setText("(m/min)");
jLabel134.setText("Maximum speed:");
jLabel137.setText("(m/min)");
jLabel139.setText("Suggested number of boxes:");
jLabel140.setText("(boxes)");
jLabel141.setText("Suggested speed");
jLabel148.setText("Requested number of boxes:");
jLabel142.setText("(boxes)");
jLabel145.setText("Suggested speed:");
jLabel146.setText("(m/min)");
jLabel149.setText("Maximum number of boxes:");
jLabel143.setText("(boxes)");
jLabel152.setText("* Conveyor Optimizer provides a heuristic and approximate solution always underestimating conveyor's capacity.");
jLabel151.setText("* Diferent or optimal solutions may exist.");
//PESTAÑA "SIMULACIÓN"
btnSimulacion.setText("Display conveyor");
jLabel147.setText("* Maximum number of boxes which fit in the conveyor based on geometric dimensions is shown.");
jLabel150.setText("* Boxes are shown in their initial position.");
//PESTAÑA "ACERCA DE"
jLabel113.setText("Author:");
jLabel115.setText("Qualification:");
jLabel116.setText("Industrial Engineer");
jLabel117.setText("Email:");
jLabel119.setText("College:");
jLabel122.setText("Tutor:");
jLabel124.setText("State:");
jLabel123.setText("Spain");
jLabel125.setText("Year:");
jLabel128.setText("* Conveyor Optimizer has been developed as a Final Thesis of Industrial Engineering degree under the German company F.EE GmbH Automation specifications.");
jLabel129.setText("* This software should be used as a complementary tool to assist in the desing and analysis of conveyors process.");
jLabel130.setText("* Neither the author nor the tutor of the Thesis are responsible for the consequences arising from its use.");
jLabel131.setText("* Its inappropriated use or results misinterpretation may lead to unknown consequences.");
}

private void btnDeutschActionPerformed(java.awt.event.ActionEvent evt) {
    idioma = 4;
    JOptionPane.showMessageDialog(null, "Sprache ist auf Deutsch eingestellt");
    /*Mensajes emergentes*/
    mensaje1 = "Alle Werte sind richtig eingetragen";
    mensaje2 = "Achtung: Bitte wählen Sie zuerst ein gültiges Datei von Fördererart";
    mensaje3 = "Achtung: Der max. Druckkraft pro Stopper muss gleich oder höher als der Druckkraft pro Palette";
    mensaje4 = "Achtung: Eingetragene Palettenabstand kleiner als min. Palettenabstand";
    mensaje5 = "Achtung: Eingetragene Beladeabstand kleiner als min. Beladeabstand";
    mensaje6 = "Achtung: Eingetragene Entnahmeabstand kleiner als min. Entnahmeabstand";
    mensaje7 = "Achtung: Abstand zwischen Belade Iniziatorposition und Beladestelleposition kleiner als 1/2 von Paletteabstand";
    mensaje8 = "Achtung: Abstand zwischen Entlade Iniziatorposition und Ende der Sperrgebiet Umlenkung auf den Rückweg kleiner als 1/2 von Paletteabstand";
    mensaje9 = "Achtung: Abstand zwischen Entladestelle und Wartestelle auf Entladen kleiner als Palettenabstand + Delta Stopper oder Robotersperrabstand beim Entladen + 1/2 Palettenabstand + Delta Stopper";
    mensaje10 = "Achtung: Abstand zwischen Nullpunkt zwischen Umlenkung beim Beladestelle und Wartestelle beim Beladen kleiner als Palettenabstand + Delta Stopper";
    mensaje11 = "Achtung: Im Förderrichtung, eingeführte Achsenabstand zu klein";
    mensaje12 = "Achtung: Im Rückrichtung, eingeführte Achsenabstand zu klein";
}

```

```

mensaje13 = "Achtung:Takzeit kleiner als Beladezeit + Entladezeit";
mensaje14 = "Bestätigung: Alle eingeführte Parameter sind kompatibel mit diesen Fördererart";
mensaje15 = "Bitte: zuerst Parameter einführen und bestätigen";
mensaje16 = "Bitte: zuerst die geometrische Ergebnisse ergattern/kalkulieren";
mensaje17 = "Achtung: Fördergeschwindigkeit zu niedriger. Keine Lösung erreicht.";
mensaje18 = "Achtung: eingeführte Geschwindigkeit kleiner als min. Geschwindigkeit";
mensaje19 = "Achtung: eingeführte Geschwindigkeit grösser als min. Geschwindigkeit";
mensaje20 = "Funktion leerlauf Förderer:";
mensaje21 = "Funktion leerlauf Förderer: JA";
mensaje22 = "Funktion leerlauf Förderer: NEIN";
mensaje23 = "Achtung: Palettenzahl kann nicht kleiner als 1 sein";
mensaje24 = "Achtung: Palettenzahl kann nicht grösser als max. Palettenzahl sein";
mensaje25 = "Hinweis: min. Fördergeschwindigkeit ist zwingen festgelegt";
mensaje26 = "Achtung: Palettenzahl zu klein. Fördergeschwindigkeit überschreitet die max. Werte";
mensaje27 = "Hinweis: gemäss resultierende Fördergeschwindigkeit sind mehr Paletten im Förderer als notwendig";

//NOMBRES PESTANAS
jTabbedPane1.setTitleAt(0, " Anfang ");
jTabbedPane1.setTitleAt(1, " Förderer Darstellung ");
jTabbedPane1.setTitleAt(2, " Fördererart ");
jTabbedPane1.setTitleAt(3, " Parameter ");
jTabbedPane1.setTitleAt(4, " Geometrische Ergebnisse ");
jTabbedPane1.setTitleAt(5, " Dinamische Ergebnisse ");
jTabbedPane1.setTitleAt(6, " Simulation ");
jTabbedPane1.setTitleAt(7, " Über ");

//PESTAÑA "ESQUEMA CINTA" Förderer Darstellung
jLabel169.setText("Stopper");
jLabel168.setText("Initiator");
//PESTAÑA "TIPO DE CINTA"
jLabel16.setText("Grundparameter des Förderers");
jLabel7.setText("Datei der Fördererart:");
btnLeerArchivoCinta.setText("Suchen");
jLabel8.setText("Max. Druckkraft pro Stopper:");
jLabel9.setText("Umlenkungsdurchmesser:");
jLabel10.setText("Min. Entladeabstand:");
jLabel11.setText("Min. Beladeabstand:");
jLabel12.setText("Min. Palettenabstand:");
jLabel13.setText("Delta Stopper:");
jLabel14.setText("Min. Fördergeschwindigkeit:");
jLabel15.setText("Max. Fördergeschwindigkeit:");
jLabel61.setText("Abstand vor Entladeumlenkung:");
jLabel64.setText("Abstand vor Beladeumlenkung:");
jLabel65.setText("Abstand nach Entladeumlenkung:");
jLabel68.setText("Abstand nach Beladeumlenkung:");
jLabel83.setText("Delta Wartungsstopper beim Beladen:");
//PESTAÑA "PARÁMETROS"
jLabel25.setText("Bitte operative Förderparameter einfügen");
jLabel26.setText("ABSTÄNDE");
jLabel28.setText("Achsenabstand:");
jLabel29.setText("Koordinate Beladestelle:");
jLabel31.setText("Koordinate Beladestelle:");
jLabel32.setText("Koordinate Beladeiniziator:");
jLabel35.setText("Koordinate Entladeiniziator:");
jLabel38.setText("Koordinate 2. Stopper auf die Förderstrecke:");
jLabel39.setText("Koordinate 1. Stopper auf die Rückstrecke:");
jLabel41.setText("FUNKTION LEERLAUF FÖRDERER");
jLabel42.setText("Taktzeit:");
jLabel44.setText("Beladezeit:");
jLabel45.setText("Entladezeit:");
jLabel46.setText("Netto Förderzeit:");
jLabel51.setText("SONTIGES");
jLabel52.setText("Palettedruckkraft:");
jLabel53.setText("Palettenabstand:");
jLabel54.setText("Sperrabstand für Beladeroboter:");

```

```

jLabel55.setText("Sperrabstand für Entladeroboter:");
jLabel60.setText("ZEIT");
chkPulmonVacio.setText("Leer Förderer");
btnValidarParametros.setText("Parameter bestätigen");
jLabel85.setText("Max. Anzahl der Paletten pro Stopper:");
jLabel86.setText("(Paletten)");
//PESTAÑA "RESULTADOS GEOMÉTRICOS"
jLabel111.setText(mensaje20);
jLabel69.setText("Ergebnisse gemäss Förderer Geometrie");
jLabel70.setText("FÖRDERRICHTUNG");
jLabel72.setText("Anzahl der Stopper:");
jLabel73.setText("Stopper Koordinaten:");
jLabel74.setText("Anzahl der Iniziatoren:");
jLabel96.setText("*Koordinaten in mm im Bezug auf Umlenkungachse");
jLabel97.setText("Koordinaten der Iniziatoren:");
jLabel88.setText("GESAMT");
jLabel89.setText("Max. Anzahl der Paletten:");
jLabel90.setText("Anzahl der Stopper:");
jLabel91.setText("Anzahl der iniziatoren:");
btnResultadosGeometricos.setText("Ergebnisse rechnen");
jLabel81.setText("Anzahl der Paletten pro Stopper:");
chkRepartirTopes_ida.setText("Stopper gleichmäßig verteilen");
ComboBox1 = "Letzte";
ComboBox2 = "Vorletzte";
ComboBxPenultim_ida.setModel(new javax.swing.DefaultComboBoxModel(new String[]{ComboBox1, ComboBox2}));
jLabel87.setText("Iniziator Palettenpresenz:");
jLabel95.setText("Palettenzahl auf den letzten Stopper:");
chkRepartirTopes_v.setText("Stopper gleichmäßig verteilen");
jLabel82.setText("Anzahl der Paletten pro Stopper:");
jLabel98.setText("Palettenzahl auf den letzten Stopper:");
jLabel75.setText("Anzahl der Stopper:");
jLabel76.setText("Anzahl der Paletten im Stau:");
jLabel77.setText("Anzahl der iniziatoren:");
jLabel99.setText("*Koordinaten in mm im Bezug auf Entladeachse");
jLabel78.setText("Stopper Koordinaten:");
jLabel79.setText("RÜCKSTRECKE");
ComboBxPenultim_v.setModel(new javax.swing.DefaultComboBoxModel(new String[]{ComboBox1, ComboBox2}));
jLabel100.setText("Koordinaten der Sensoren:");
jLabel92.setText("Iniziator Palettenpresenz:");
jLabel80.setText("Achsenabstand:");
jLabel94.setText("Palettenabstand:");
jLabel101.setText("Anzahl der Paletten im Stau:");
jLabel103.setText("Anzahl der Paletten beim Entladen:");
jLabel104.setText("Anzahl der Paletten beim Beladen:");
jLabel106.setText("Gesamt Palettenzahl:");
jLabel108.setText("Max. Anzahl der Paletten pro Stopper:");
jLabel109.setText("Max. Anzahl der Paletten pro Stopper:");
//PESTAÑA "RESULTADOS DINÁMICOS"
btnVN.setText("Rechnen");
btnNV.setText("Rechnen");
jLabel127.setText("Vorgeschlagene Anzahl der Paletten");
jLabel132.setText("Min. Fördergeschwindigkeit:");
jLabel135.setText("(m/min)");
jLabel133.setText("Gewünschte Fördergeschwindigkeit:");
jLabel136.setText("(m/min)");
jLabel134.setText("Min. Fördergeschwindigkeit:");
jLabel137.setText("(m/min)");
jLabel139.setText("Vorgeschlagene Anzahl der Paletten:");
jLabel140.setText("(paletten)");
jLabel141.setText("Vorgeschlagene Fördergeschwindigkeit");
jLabel148.setText("Gewünschte Anzahl der Paletten:");
jLabel142.setText("Paletten");

```

```

jLabel145.setText("Vorgeschlagene Fördergeschwindigkeit:");
jLabel146.setText("(m/min)");
jLabel149.setText("Max. Anzahl der Paletten:");
jLabel143.setText("(Paletten)");
jLabel152.setText("* Conveyor Optimizer schlägt eine heuristische und abgeschätzte Lösung, die immer die Kapazität des Förderers unterschätzt.");
jLabel151.setText("* Können auch anderen Lösungen als die bereits vorgeschlagenen möglich sein.");
//PESTAÑA "SIMULACIÓN"
btnSimulacion.setText("Förderer darstellen");
jLabel147.setText("* Max. Anzahl der Paletten gemäss die geometrische Abmessungen des Förderers dargestellt.");
jLabel150.setText("* Die Paletten sind auf die Start-position dargestellt.");
//PESTAÑA "ACERCA DE"
jLabel113.setText("Verfasser:");
jLabel115.setText("Titel:");
jLabel116.setText("Diplomm-Ingenieur");
jLabel117.setText("Email:");
jLabel119.setText("Universität:");
jLabel122.setText("Coach:");
jLabel124.setText("Land:");
jLabel123.setText("Spanien");
jLabel125.setText("Jahr:");
jLabel128.setText("* Conveyor Optimizer ist als Endarbeit des Studiums Diplomm-Ingenieur abgewickelt und gemäss technische Anforderungen von der Deutsche Firma F.EE GmbH Automation.");
jLabel129.setText("* Diesers Programms ist als Ergänzungswerkzeug für die Konstruktion, Festlegung und Analise von Förderer.");
jLabel130.setText("* Verfasser und Coach machen übernehmen keine Verantwortung, die auf die Nutzung dieses Programms sich ergeben können.");
jLabel131.setText("* Die Fallsche Einsetzung bzw. fallsche Interpretation der ergebnisse können unbekannten Konsequenzen tragen.");
}

private void btnVNActionPerformed(java.awt.event.ActionEvent evt) {
    /*Borrar el jTextField de resultado anterior para evitar confusiones*/
    jTextField54.setText("");
    jTextField59.setText("");
    /*Comprobar si se han obtenido Resultados Geometricos correctamente*/
    if (!ResGeom) {
        /*Si no se han obtenido, se muestra un mensaje y se detiene el programa*/
        JOptionPane.showMessageDialog(null, mensaje16);
        try {
            this.wait();
        } catch (InterruptedException ex) {
            Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
        }
    } else {
    }

    // VELOCIDAD
    // Leer la velocidad deseada:
    double v = Double.parseDouble(jTextField49.getText()); // Velocidad en [m/min]
    // Escribir la velocidad en rpm en su jTextField
    jTextField59.setText(String.valueOf((float) (v * (1000. / (Math.PI * D)))));

    // Comprobar que (vmin <= v <= vmax):
    if (v < v_min) {
        /*Si v es inferior a la vmin, detener la ejecución
        JOptionPane.showMessageDialog(null, mensaje18);
        try {
            this.wait();
        } catch (InterruptedException ex) {
            Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
        }
    } else if (v > v_max) {
}
}

```

```

/*Si v es superior a la vmax, detener la ejecución
JOptionPane.showMessageDialog(null, mensaje19);
try {
    this.wait();
} catch (InterruptedException ex) {
    Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
}
}
v = v * (1000.0 / 60.0); //Velocidad en [mm/s]

// INICIO ALGORITMO V-N
// Tiempos que se tardan:
double tA = ((L - dG_ida - dCar) - (dEG_ida + dDesc)) / v; // t de carga a descarga
double tB = dDesc / v;
double tC = (L - dT_v01 - dEG_v - dG_v) / v;
double tD = dCar / v;
double tw = (Math.PI * D) / (2 * v);
double alpha1 = Math.atan(2 * dEG_v / D);
double alpha2 = Math.atan(2 * dG_ida / D);
double tgirar_ida = ((alpha1 + Math.PI + alpha2) * D) / (2 * v);
double tgiro_ida = tw + tgirar_ida;
double alpha3 = Math.atan(2 * dEG_ida / D);
double alpha4 = Math.atan(2 * dG_v / D);
double tgirar_v = ((alpha3 + Math.PI + alpha4) * D) / (2 * v);
double tgiro_v = tw + tgirar_v;

// Algoritmo:
boolean end = false;
boolean EROR = false;
int Nmax = Nmax_TOT;
int N = 1;
// Empezamos comprobando paleta 1 (la de posición de carga):
double tarda1 = tCarga + tA + tDescarga + tB + tgiro_v + tC + dT_v01 / v + tgiro_ida + tD;
if (tarda1 <= tCiclo) {
    end = true; // con 1 paleta: OK
} else if (tarda1 > tCiclo) {
    N = N + 1; // añadimos paleta 2 (en posición de descarga)
    if (N > Nmax) { // NO podemos añadir más paletas-> NO hay solución
        EROR = true;
        end = true;
        /* se acaba el algoritmo pq NO hay solución con esta velocidad
         * (se necesitarían más paletas de las que caben) */
    }
    // comprobamos paleta 1 (la de carga):
    tarda1 = tCarga + tA;
    if ((tarda1 > tCiclo) & (!end)) { // si falla por la 1
        N = N + 1; // se añade paleta en el tope T_ida1
        if (N > Nmax) {
            EROR = true;
            end = true;
        }
        // comprobamos paleta del tope T_ida1
        double tarda1cola = dT_ida01 / v;
        if ((tarda1cola > tCiclo - tDescarga) & (!end)) {
            EROR = true;
            end = true;
            //NO hay solución para esta velocidad
        }
        // comprobamos paleta 1 (la de carga):
        int cont = 1;
        tarda1 = tCarga + tA - dT_ida01 / v;
        while ((tarda1 > tCiclo - tDescarga) & (!end)) {
            N = N + 1; // se añade paleta a continuación de las del tope T_ida1
            if (N > Nmax) {

```

```

        EROR = true;
        end = true;
    }
    tarda1 = tCarga + tA - dT_ida01 / v - cont * pas;
    /* iteramos: se van acumulando paletas y se negligen los delta
     * tope para hacerlo más restrictivo y pq (delta<<pas)*/
    cont = cont + 1;
}
}
// comprobamos paleta 2 (la de descarga):
double tarda2 = tDescarga + tB + tgiro_v + tC + dT_v01 / v + tgiro_ida + tD;
if ((tarda2 > tCiclo) & (!end)) { // si falla por la 2
    N = N + 1; // añadimos paleta 3 (en el tope T_v1)
    if (N > Nmax) {
        EROR = true;
        end = true;
    }
    // comprobamos paleta 3 (la del tope T_v1):
    double tarda3 = dT_v01 / v + tgiro_ida + tD;
    if ((tarda3 > tCiclo - tCarga) & (!end)) {
        EROR = true;
        end = true;
        // si (tarda3>tCiclo-tCarga): NO hay solución para esta velocidad
    }
    // comprobamos paleta 2 (la de descarga):
    tarda2 = tarda2 - tD - tgiro_ida - dT_v01 / v;
    int cont = 1;
    while ((tarda2 > tCiclo - tCarga) & (!end)) {
        N = N + 1; // se van añadiendo paletas detrás de la del T_v1
        if (N > Nmax) {
            EROR = true;
            end = true;
        }
        tarda2 = tarda2 - cont * pas;
        // tiene que poder llegar hasta la posición de la paleta que se va
        cont = cont + 1;
    }
}
}
// FIN ALGORITMO V-N

// Mostrar resultados
if (EROR) {
    JOptionPane.showMessageDialog(null, mensaje17);
    jTextField54.setText("ERROR");
} else if (!EROR) {
    jTextField54.setText(String.valueOf(N));
}
}

private void btnNVActionPerformed(java.awt.event.ActionEvent evt) {
    /*Borrar el jTextField de resultado anterior para evitar confusiones*/
    jTextField55.setText("");
    jTextField61.setText("");
    /*Comprobar si se han obtenido Resultados Geometricos correctamente*/
    if (!ResGeom) {
        /*Si no se han obtenido, se muestra un mensaje y se detiene el programa*/
        JOptionPane.showMessageDialog(null, mensaje16);
        try {
            this.wait();
        } catch (InterruptedException ex) {
            Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
        }
    } else {
}
}

```

```

}

// NÚMERO DE PALETAS DESEADO
// Leer el número de paletas deseado:
int Ndeseado = Integer.parseInt(jTextField56.getText());
// Comprobar que (1 <= Ndeseado <= Nmax):
int Nmax = Nmax_TOT;
if (Ndeseado < 1) {
    /*Si Ndeseado es inferior a 1, detener la ejecución
    JOptionPane.showMessageDialog(null, mensaje23);
    try {
        this.wait();
    } catch (InterruptedException ex) {
        Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
    }
} else if (Ndeseado > Nmax) {
    /*Si Ndeseado es superior a Nmax, detener la ejecución
    JOptionPane.showMessageDialog(null, mensaje24);
    try {
        this.wait();
    } catch (InterruptedException ex) {
        Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
    }
}

// INICIO ALGORITMO OPTIMIZACIÓN
// Partimos de v = vmin
double v = v_min; // [m/min]
v = v * (1000.0 / 60.0); // Velocidad en [mm/s]

// Algoritmo V-N
// Tiempos que se tardan:
double tA = ((L - dG_ida - dCar) - (dEG_ida + dDesc)) / v; // t de carga a descarga
double tB = dDesc / v;
double tC = (L - dT_v01 - dEG_v - dG_v) / v;
double tD = dCar / v;
double tw = (Math.PI * D) / (2 * v);
double alpha1 = Math.atan(2 * dEG_v / D);
double alpha2 = Math.atan(2 * dG_ida / D);
double tgirar_ida = ((alpha1 + Math.PI + alpha2) * D) / (2 * v);
double tgiro_ida = tw + tgirar_ida;
double alpha3 = Math.atan(2 * dEG_ida / D);
double alpha4 = Math.atan(2 * dG_v / D);
double tgirar_v = ((alpha3 + Math.PI + alpha4) * D) / (2 * v);
double tgiro_v = tw + tgirar_v;

boolean end = false;
int N = 1;
// Empezamos comprobando paleta 1 (la de posición de carga):
double tarda1 = tCarga + tA + tDescarga + tB + tgiro_v + tC + dT_v01 / v + tgiro_ida + tD;
if (tarda1 <= tCiclo) {
    end = true; // con 1 paleta: OK
} else if (tarda1 > tCiclo) {
    N = N + 1; // añadimos paleta 2 (en posición de descarga)
    if (N > Nmax) {
        end = true;
        /* se acaba el algoritmo pq NO hay solución con esta velocidad
        * (se necesitarían más paletas de las que caben)*/
    }
    // comprobamos paleta 1 (la de carga):
    tarda1 = tCarga + tA;
    if ((tarda1 > tCiclo) & (!end)) { // si falla por la 1
        N = N + 1; // se añade paleta en el tope T_ida1
        if (N > Nmax) {
}

```

```

        end = true;
    }
    // comprobamos paleta del tope T_ida1
    double tarda1cola = dT_ida01 / v;
    if ((tarda1cola > tCiclo - tDescarga) & (!end)) {
        end = true;
        //NO hay solución para esta velocidad
    }
    //comprobamos paleta 1 (la de carga):
    int cont = 1;
    tarda1 = tCarga + tA - dT_ida01 / v;
    while ((tarda1 > tCiclo - tDescarga) & (!end)) {
        N = N + 1; //se añade paleta a continuación de las del tope T_ida1
        if (N > Nmax) {
            end = true;
        }
        tarda1 = tCarga + tA - dT_ida01 / v - cont * pas;
        /* iteramos: se van acumulando paletas y se negligen los delta
         * tope para hacerlo más restrictivo y pq (delta<<pas)*/
        cont = cont + 1;
    }
}
// comprobamos paleta 2 (la de descarga):
double tarda2 = tDescarga + tB + tgiro_v + tC + dT_v01 / v + tgiro_ida + tD;
if ((tarda2 > tCiclo) & (!end)) { // si falla por la 2
    N = N + 1; // añadimos paleta 3 (en el tope T_v1)
    if (N > Nmax) {
        end = true;
    }
    // comprobamos paleta 3 (la del tope T_v1):
    double tarda3 = dT_v01 / v + tgiro_ida + tD;
    if ((tarda3 > tCiclo / tCarga) & (!end)) {
        end = true;
        // si (tarda3>tCiclo/tCarga): NO hay solución para esta velocidad
    }
    // comprobamos paleta 2 (la de descarga):
    tarda2 = tarda2 - tD - tgiro_ida - dT_v01 / v;
    int cont = 1;
    while ((tarda2 > tCiclo - tCarga) & (!end)) {
        N = N + 1; // se van añadiendo paletas detrás de la del T_v1
        if (N > Nmax) {
            end = true;
        }
        tarda2 = tarda2 - cont * pas;
        // tiene que poder llegar hasta la posición de la paleta que se va
        cont = cont + 1;
    }
}
}
// Fin algoritmo V-N. Obtenemos N

/* Si Nobtenido < Ndeseado, significa que usamos la velocidad mínima
 * (y no entramos en el while)*/
if (N < Ndeseado) {
    JOptionPane.showMessageDialog(null, mensaje25);
}
// Mientras Nobtenido > Ndeseado, iteramos aumentando la velocidad
while ((N > Ndeseado)) {
    /* Aumentamos la v en 0.1 m/min = 0.1*1000/60 mm/s y comprobamos
     * si hemos pasado la máxima:*/
    v = v + 0.1 * (1000.0 / 60.0);
    if (v > (v_max * (1000.0 / 60.0))) {
        //Si pasamos de la v_max, no hay solución, detener ejecución
        jTextField55.setText("ERROR");
    }
}

```

```

jTextField61.setText("ERROR");
JOptionPane.showMessageDialog(null, mensaje26);
try {
    this.wait();
} catch (InterruptedException ex) {
    Logger.getLogger(Interface.class.getName()).log(Level.SEVERE, null, ex);
}
}

// Algoritmo V-N
// Tiempos que se tardan:
tA = ((L - dG_ida - dCar) - (dEG_ida + dDesc)) / v;
tB = dDesc / v;
tC = (L - dT_v01 - dEG_v - dG_v) / v;
tD = dCar / v;
tw = (Math.PI * D) / (2 * v);
alpha1 = Math.atan(2 * dEG_v / D);
alpha2 = Math.atan(2 * dG_ida / D);
tgirar_ida = ((alpha1 + Math.PI + alpha2) * D) / (2 * v);
tgiro_ida = tw + tgirar_ida;
alpha3 = Math.atan(2 * dEG_ida / D);
alpha4 = Math.atan(2 * dG_v / D);
tgirar_v = ((alpha3 + Math.PI + alpha4) * D) / (2 * v);
tgiro_v = tw + tgirar_v;

end = false;
N = 1;
// Empezamos comprobando paleta 1 (la de posición de carga):
tarda1 = tCarga + tA + tDescarga + tB + tgiro_v + tC + dT_v01 / v + tgiro_ida + tD;
if (tarda1 <= tCiclo) {
    end = true; // con 1 paleta: OK
} else if (tarda1 > tCiclo) {
    N = N + 1; // añadimos paleta 2 (en posición de descarga)
    if (N > Nmax) {
        end = true;
        /* se acaba el algoritmo pq NO hay solución con esta
         * velocidad (se necesitarían más paletas de las que caben)*/
    }
    // comprobamos paleta 1 (la de carga):
    tarda1 = tCarga + tA;
    if ((tarda1 > tCiclo) & (!end)) { // si falla por la 1
        N = N + 1; // se añade paleta en el tope T_ida1
        if (N > Nmax) {
            end = true;
        }
        // comprobamos paleta del tope T_ida1
        double tarda1cola = dT_ida01 / v;
        if ((tarda1cola > tCiclo - tDescarga) & (!end)) {
            end = true;
            //NO hay solución para esta velocidad
        }
        // comprobamos paleta 1 (la de carga):
        int cont = 1;
        tarda1 = tCarga + tA - dT_ida01 / v;
        while ((tarda1 > tCiclo - tDescarga) & (!end)) {
            N = N + 1; //se añade paleta a continuación de las del tope T_ida1
            if (N > Nmax) {
                end = true;
            }
            tarda1 = tCarga + tA - dT_ida01 / v - cont * pas;
            /* se van acumulando paletas y se negligen los delta tope
             * para hacerlo más restrictivo y pq (delta<<<pas)*/
            cont = cont + 1;
        }
    }
}

```

```

        }
        // comprobamos paleta 2 (la de descarga):
        double tarda2 = tDescarga + tB + tgiro_v + tC + dT_v01 / v + tgiro_ida + tD;
        if ((tarda2 > tCiclo) & (!end)) { // si falla por la 2
            N = N + 1; // añadimos paleta 3 (en el tope T_v1)
            if (N > Nmax) {
                end = true;
            }
            // comprobamos paleta 3 (la del tope T_v1):
            double tarda3 = dT_v01 / v + tgiro_ida + tD;
            if ((tarda3 > tCiclo - tCarga) & (!end)) {
                end = true;
                //si (tarda3>tCiclo-tCarga): NO hay solución para esta velocidad
            }
            // comprobamos paleta 2 (la de descarga):
            tarda2 = tarda2 - tD - tgiro_ida - dT_v01 / v;
            int cont = 1;
            while ((tarda2 > tCiclo - tCarga) & (!end)) {
                N = N + 1; // se van añadiendo paletas detras de la del T_v1
                if (N > Nmax) {
                    end = true;
                }
                tarda2 = tarda2 - cont * pas;
                //tiene que poder llegar hasta la posición de la paleta que se va
                cont = cont + 1;
            }
        }
    }
    // Fin algoritmo V-N. Obtenemos N
}
// FIN ALGORITMO OPTIMIZACIÓN

// Resultados
// Mostrar resultados
jTextField55.setText(String.valueOf((float) (v * (60. / 1000.))));
jTextField61.setText(String.valueOf((float) (v * (60 / (Math.PI * D)))));
// Si Nobtenido < Ndeseado, hay paletas de más
if (N < Ndeseado) {
    JOptionPane.showMessageDialog(null, mensaje27);
}
}

/**
 * @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(Interface.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    } catch (InstantiationException ex) {
        java.util.logging.Logger.getLogger(Interface.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    }
}

```

```
    } catch (IllegalAccessException ex) {
        java.util.logging.Logger.getLogger(Interface.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {
        java.util.logging.Logger.getLogger(Interface.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
    }
//</editor-fold>

/* Create and display the form */
java.awt.EventQueue.invokeLater(new Runnable() {
    public void run() {
        new Interface().setVisible(true);
    }
});
}

// Variables declaration - do not modify
private javax.swing.JComboBox ComboBxPenultim_ida;
private javax.swing.JComboBox ComboBxPenultim_v;
private javax.swing.JButton btnCatala;
private javax.swing.JButton btnDeutsch;
private javax.swing.JButton btnEnglish;
private javax.swing.JButton btnEspanol;
private javax.swing.JButton btnLeerArchivoCinta;
private javax.swing.JButton btnNV;
private javax.swing.JButton btnResultadosGeometricos;
private javax.swing.JButton btnSimulacion;
private javax.swing.JButton btnVN;
private javax.swing.JButton btnValidarParametros;
private javax.swing.JCheckBox chkPulmonVacio;
private javax.swing.JCheckBox chkRepartirTopes_ida;
private javax.swing.JCheckBox chkRepartirTopes_v;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel10;
private javax.swing.JLabel jLabel100;
private javax.swing.JLabel jLabel101;
private javax.swing.JLabel jLabel102;
private javax.swing.JLabel jLabel103;
private javax.swing.JLabel jLabel104;
private javax.swing.JLabel jLabel105;
private javax.swing.JLabel jLabel106;
private javax.swing.JLabel jLabel107;
private javax.swing.JLabel jLabel108;
private javax.swing.JLabel jLabel109;
private javax.swing.JLabel jLabel11;
private javax.swing.JLabel jLabel110;
private javax.swing.JLabel jLabel111;
private javax.swing.JLabel jLabel112;
private javax.swing.JLabel jLabel113;
private javax.swing.JLabel jLabel114;
private javax.swing.JLabel jLabel115;
private javax.swing.JLabel jLabel116;
private javax.swing.JLabel jLabel117;
private javax.swing.JLabel jLabel118;
private javax.swing.JLabel jLabel119;
private javax.swing.JLabel jLabel12;
private javax.swing.JLabel jLabel120;
private javax.swing.JLabel jLabel121;
private javax.swing.JLabel jLabel122;
private javax.swing.JLabel jLabel123;
private javax.swing.JLabel jLabel124;
private javax.swing.JLabel jLabel125;
private javax.swing.JLabel jLabel126;
private javax.swing.JLabel jLabel127;
```

```
private javax.swing.JLabel jLabel128;
private javax.swing.JLabel jLabel129;
private javax.swing.JLabel jLabel13;
private javax.swing.JLabel jLabel130;
private javax.swing.JLabel jLabel131;
private javax.swing.JLabel jLabel132;
private javax.swing.JLabel jLabel133;
private javax.swing.JLabel jLabel134;
private javax.swing.JLabel jLabel135;
private javax.swing.JLabel jLabel136;
private javax.swing.JLabel jLabel137;
private javax.swing.JLabel jLabel138;
private javax.swing.JLabel jLabel139;
private javax.swing.JLabel jLabel14;
private javax.swing.JLabel jLabel140;
private javax.swing.JLabel jLabel141;
private javax.swing.JLabel jLabel142;
private javax.swing.JLabel jLabel143;
private javax.swing.JLabel jLabel144;
private javax.swing.JLabel jLabel145;
private javax.swing.JLabel jLabel146;
private javax.swing.JLabel jLabel147;
private javax.swing.JLabel jLabel148;
private javax.swing.JLabel jLabel149;
private javax.swing.JLabel jLabel15;
private javax.swing.JLabel jLabel150;
private javax.swing.JLabel jLabel151;
private javax.swing.JLabel jLabel152;
private javax.swing.JLabel jLabel153;
private javax.swing.JLabel jLabel154;
private javax.swing.JLabel jLabel155;
private javax.swing.JLabel jLabel156;
private javax.swing.JLabel jLabel157;
private javax.swing.JLabel jLabel158;
private javax.swing.JLabel jLabel159;
private javax.swing.JLabel jLabel16;
private javax.swing.JLabel jLabel160;
private javax.swing.JLabel jLabel161;
private javax.swing.JLabel jLabel162;
private javax.swing.JLabel jLabel163;
private javax.swing.JLabel jLabel164;
private javax.swing.JLabel jLabel165;
private javax.swing.JLabel jLabel166;
private javax.swing.JLabel jLabel167;
private javax.swing.JLabel jLabel168;
private javax.swing.JLabel jLabel169;
private javax.swing.JLabel jLabel17;
private javax.swing.JLabel jLabel18;
private javax.swing.JLabel jLabel19;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel20;
private javax.swing.JLabel jLabel21;
private javax.swing.JLabel jLabel22;
private javax.swing.JLabel jLabel23;
private javax.swing.JLabel jLabel24;
private javax.swing.JLabel jLabel25;
private javax.swing.JLabel jLabel26;
private javax.swing.JLabel jLabel27;
private javax.swing.JLabel jLabel28;
private javax.swing.JLabel jLabel29;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel30;
private javax.swing.JLabel jLabel31;
private javax.swing.JLabel jLabel32;
```

```
private javax.swing.JLabel jLabel33;
private javax.swing.JLabel jLabel34;
private javax.swing.JLabel jLabel35;
private javax.swing.JLabel jLabel36;
private javax.swing.JLabel jLabel37;
private javax.swing.JLabel jLabel38;
private javax.swing.JLabel jLabel39;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel40;
private javax.swing.JLabel jLabel41;
private javax.swing.JLabel jLabel42;
private javax.swing.JLabel jLabel43;
private javax.swing.JLabel jLabel44;
private javax.swing.JLabel jLabel45;
private javax.swing.JLabel jLabel46;
private javax.swing.JLabel jLabel47;
private javax.swing.JLabel jLabel48;
private javax.swing.JLabel jLabel49;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel50;
private javax.swing.JLabel jLabel51;
private javax.swing.JLabel jLabel52;
private javax.swing.JLabel jLabel53;
private javax.swing.JLabel jLabel54;
private javax.swing.JLabel jLabel55;
private javax.swing.JLabel jLabel56;
private javax.swing.JLabel jLabel57;
private javax.swing.JLabel jLabel58;
private javax.swing.JLabel jLabel59;
private javax.swing.JLabel jLabel6;
private javax.swing.JLabel jLabel60;
private javax.swing.JLabel jLabel61;
private javax.swing.JLabel jLabel62;
private javax.swing.JLabel jLabel63;
private javax.swing.JLabel jLabel64;
private javax.swing.JLabel jLabel65;
private javax.swing.JLabel jLabel66;
private javax.swing.JLabel jLabel67;
private javax.swing.JLabel jLabel68;
private javax.swing.JLabel jLabel69;
private javax.swing.JLabel jLabel7;
private javax.swing.JLabel jLabel70;
private javax.swing.JLabel jLabel71;
private javax.swing.JLabel jLabel72;
private javax.swing.JLabel jLabel73;
private javax.swing.JLabel jLabel74;
private javax.swing.JLabel jLabel75;
private javax.swing.JLabel jLabel76;
private javax.swing.JLabel jLabel77;
private javax.swing.JLabel jLabel78;
private javax.swing.JLabel jLabel79;
private javax.swing.JLabel jLabel8;
private javax.swing.JLabel jLabel80;
private javax.swing.JLabel jLabel81;
private javax.swing.JLabel jLabel82;
private javax.swing.JLabel jLabel83;
private javax.swing.JLabel jLabel84;
private javax.swing.JLabel jLabel85;
private javax.swing.JLabel jLabel86;
private javax.swing.JLabel jLabel87;
private javax.swing.JLabel jLabel88;
private javax.swing.JLabel jLabel89;
private javax.swing.JLabel jLabel9;
private javax.swing.JLabel jLabel90;
```

```
private javax.swing.JLabel jLabel91;
private javax.swing.JLabel jLabel92;
private javax.swing.JLabel jLabel93;
private javax.swing.JLabel jLabel94;
private javax.swing.JLabel jLabel95;
private javax.swing.JLabel jLabel96;
private javax.swing.JLabel jLabel97;
private javax.swing.JLabel jLabel98;
private javax.swing.JLabel jLabel99;
private javax.swing.JPanel jPanel1;
private javax.swing.JPanel jPanel2;
private javax.swing.JPanel jPanel3;
private javax.swing.JPanel jPanel4;
private javax.swing.JPanel jPanel5;
private javax.swing.JPanel jPanel6;
private javax.swing.JPanel jPanel7;
private javax.swing.JPanel jPanel8;
private javax.swing.JTabbedPane jTabbedPane1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField10;
private javax.swing.JTextField jTextField11;
private javax.swing.JTextField jTextField12;
private javax.swing.JTextField jTextField13;
private javax.swing.JTextField jTextField14;
private javax.swing.JTextField jTextField15;
private javax.swing.JTextField jTextField16;
private javax.swing.JTextField jTextField17;
private javax.swing.JTextField jTextField18;
private javax.swing.JTextField jTextField19;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField20;
private javax.swing.JTextField jTextField21;
private javax.swing.JTextField jTextField22;
private javax.swing.JTextField jTextField23;
private javax.swing.JTextField jTextField24;
private javax.swing.JTextField jTextField25;
private javax.swing.JTextField jTextField26;
private javax.swing.JTextField jTextField27;
private javax.swing.JTextField jTextField28;
private javax.swing.JTextField jTextField29;
private javax.swing.JTextField jTextField3;
private javax.swing.JTextField jTextField30;
private javax.swing.JTextField jTextField31;
private javax.swing.JTextField jTextField32;
private javax.swing.JTextField jTextField33;
private javax.swing.JTextField jTextField34;
private javax.swing.JTextField jTextField35;
private javax.swing.JTextField jTextField36;
private javax.swing.JTextField jTextField37;
private javax.swing.JTextField jTextField38;
private javax.swing.JTextField jTextField39;
private javax.swing.JTextField jTextField4;
private javax.swing.JTextField jTextField40;
private javax.swing.JTextField jTextField41;
private javax.swing.JTextField jTextField42;
private javax.swing.JTextField jTextField43;
private javax.swing.JTextField jTextField44;
private javax.swing.JTextField jTextField45;
private javax.swing.JTextField jTextField46;
private javax.swing.JTextField jTextField47;
private javax.swing.JTextField jTextField48;
private javax.swing.JTextField jTextField49;
private javax.swing.JTextField jTextField5;
private javax.swing.JTextField jTextField50;
```

```
private javax.swing.JTextField jTextField51;
private javax.swing.JTextField jTextField52;
private javax.swing.JTextField jTextField53;
private javax.swing.JTextField jTextField54;
private javax.swing.JTextField jTextField55;
private javax.swing.JTextField jTextField56;
private javax.swing.JTextField jTextField57;
private javax.swing.JTextField jTextField58;
private javax.swing.JTextField jTextField59;
private javax.swing.JTextField jTextField6;
private javax.swing.JTextField jTextField60;
private javax.swing.JTextField jTextField61;
private javax.swing.JTextField jTextField7;
private javax.swing.JTextField jTextField8;
private javax.swing.JTextField jTextField9;
// End of variables declaration
}
```

```

package conveyoroptimizer;

/*Importamos java.io.* para poder usar la clase File y sus funciones*/
/*Importamos javax.swing.JOptionPane para poder usar el Pane emergente*/
import java.io.*;
import javax.swing.JOptionPane;

public class UsarArchivo {

    public double LeerValor(String dato, String path, int idioma) throws InterruptedException {
        // Idioma mensajes:
        String mensaje1 = "";
        String mensaje2 = "";
        if (idioma == 1) {
            mensaje1 = "" no existe o está mal escrito";
            mensaje2 = "" no es un valor válido";
        } else if (idioma == 2) {
            mensaje1 = "" no existeix o està malament escrit";
            mensaje2 = "" no és un valor vàlid";
        } else if (idioma == 3) {
            mensaje1 = "" either does not exist or it is misspelled";
            mensaje2 = "" is not a valid value";
        } else if (idioma == 4) {
            mensaje1 = "" gibt's es nicht oder fallsch geschrieben";
            mensaje2 = "" kein gültiges Wert";
        }
        /* Esta función lee, en el archivo path, el numero de debajo de la
         * linea del dato pedido*/
        File f;
        String valor_leido = "";
        double valor = 0;
        f = new File(path);
        try {
            FileReader fr = new FileReader(f);
            BufferedReader br = new BufferedReader(fr);

            String aux = br.readLine();
            while ((aux != null) & (!dato.equals(aux))) {
                aux = br.readLine();
            }
            if (aux == null) {
                JOptionPane.showMessageDialog(null, "" + dato + mensaje1);
                /*Si no está el dato, detenemos el programa*/
                this.wait();
            } else {
                valor_leido = br.readLine();
                /*Comprobamos que el valor leído sea un número*/
                try {
                    Double.parseDouble(valor_leido);
                    valor = Double.parseDouble(valor_leido);
                } catch (NumberFormatException exc) {
                    /*Si no es un número, detenemos el programa*/
                    JOptionPane.showMessageDialog(null, "" + dato + mensaje2);
                    this.wait();
                }
            }
        } catch (IOException e) {
            System.out.println("Error:" + e.getMessage());
        }
        return valor;
    }
}

```

```
package conveyoroptimizer;
import java.awt.Graphics;
public class Dibujar {
    public static void Redonda(Graphics g, int x, int y, int Diametro) {
        g.drawOval(x, y, Diametro, Diametro);
    }
    public static void Linea(Graphics g, int x1, int y1, int x2, int y2) {
        g.drawLine(x1, y1, x2, y2);
    }
    public static void PaletaArriba(Graphics g, double x, double y, double paso) {
        double aureo = 1 / 1.618;
        int a1 = (int) (x - paso / 2);
        int b1 = (int) y - (int) (paso * aureo);
        g.drawRect(a1, b1, (int) paso, (int) (paso * aureo));
    }
    public static void PaletaAbajo(Graphics g, double x, double y, double paso) {
        double aureo = 1 / 1.618;
        int a1 = (int) (x - paso / 2);
        int b1 = (int) y;
        g.drawRect(a1, b1, (int) paso, (int) (paso * aureo));
    }
}
```