

Prueba de Caja Blanca

“Sistema de Inventario”

Integrantes:

**Robinson Estrella
Fenix Toapanta
Mathias Tapia**

Fecha 2025 / 06 / 25

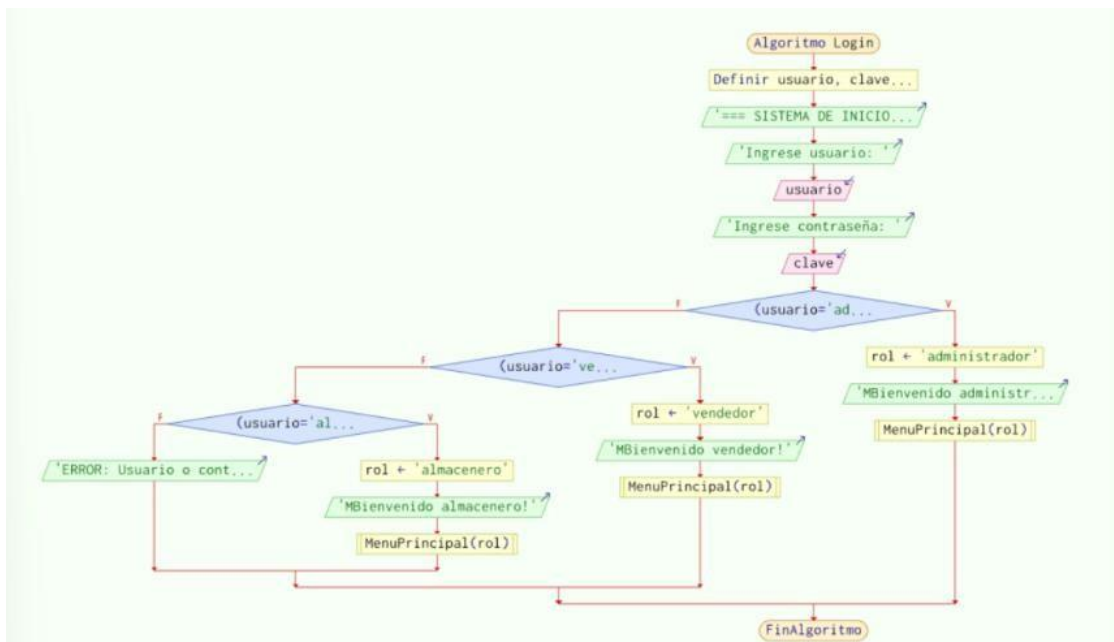
Prueba caja blanca de Inicio de sesión.

1. CÓDIGO FUENTE

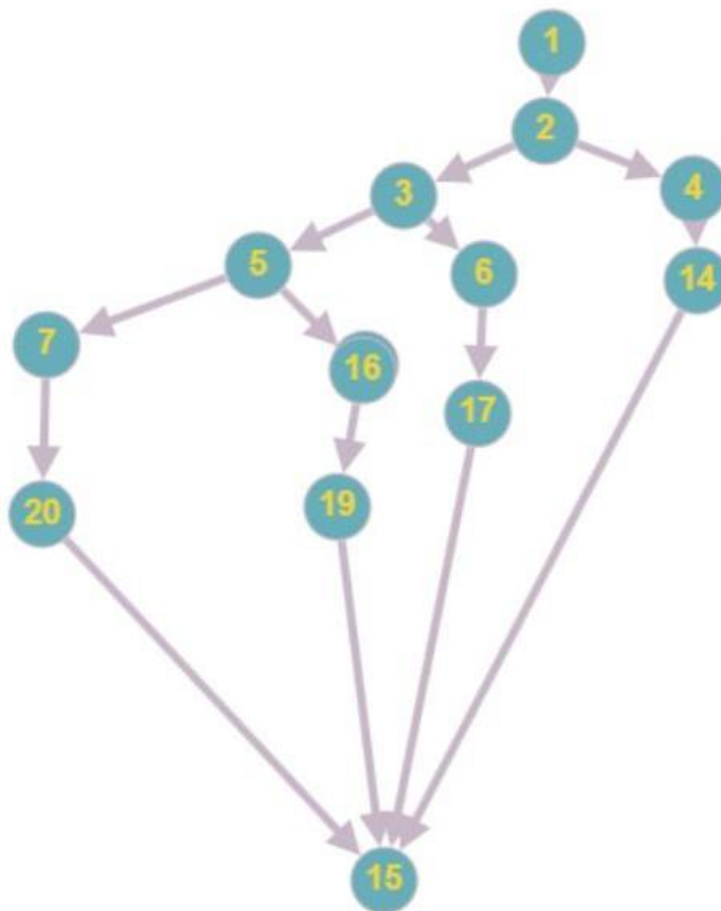
```
if(us.getUsuario()!=null && us.getPassword()!=null)
{
    MenuPrincipal m=new MenuPrincipal();
    //Administrador Vendedor Almacenero
    if(us.getTipoUsuario().equals("Vendedor")){...9 lines }else if(us.getTipoUsuario().equals("Alma
    MenuPrincipal.txtidUser.setText(us.getIdusuario()+"");
    MenuPrincipal.txtuser.setText(us.getUsuario());
    Categorias.iduser=us.getIdusuario();
    Notification panel = new Notification(this, Notification.Type.SUCCESS, Notification.Location.TOP);
    panel.showNotification();
    m.setVisible(true);
    dispose();
}else{
    //JOptionPane.showMessageDialog(null, "Aceso denegado");
    Notification panel = new Notification(this, Notification.Type.ERROR, Notification.Location.TOP);
    panel.showNotification();
}
}

private void btnVerPasswordActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    if(btnVerPassword.isSelected()){
        txtpassword.setEchoChar((char)0);
    }else{
        txtpassword.setEchoChar('*');
    }
}
```

2. DIAGRAMA DE FLUJO (DF)



3. GRAFO DE FLUJO (GF)



4. IDENTIFICACIÓN DE LAS RUTAS (Camino basico)

RUTAS

- R1:** 1 , 2 , 4 , 14 , 15
- R2:** 1 , 2 , 3 , 6 , 17 , 15
- R3:** 1 , 2 , 3 , 5 , 16 , 19 , 15
- R4:** 1 , 2 , 3 , 5 , 7 , 20 , 15

5. COMPLEJIDAD CICLOMÁTICA

Se puede calcular de las siguientes formas:

- $V(G) = \text{número de nodos predicados(decisiones)} + 1$
 $V(G) = 3 + 1 = 4$
- $V(G) = A - N + 2$
 $V(G) = 15 - 13 + 2 = 4$

DONDE:

P: Número de nodos predicado

A: Número de aristas

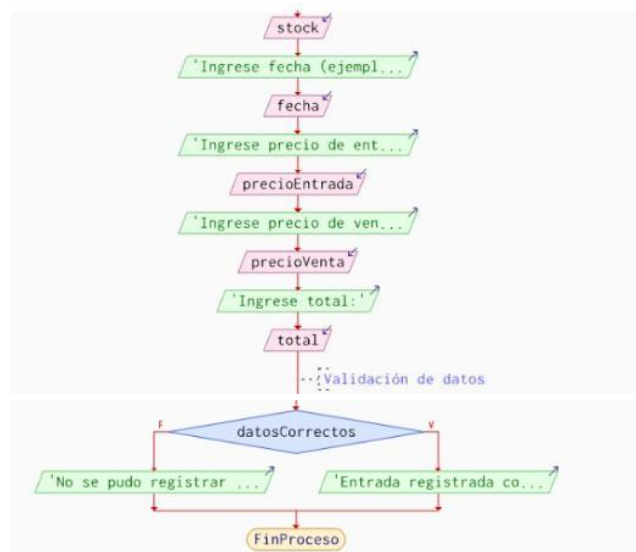
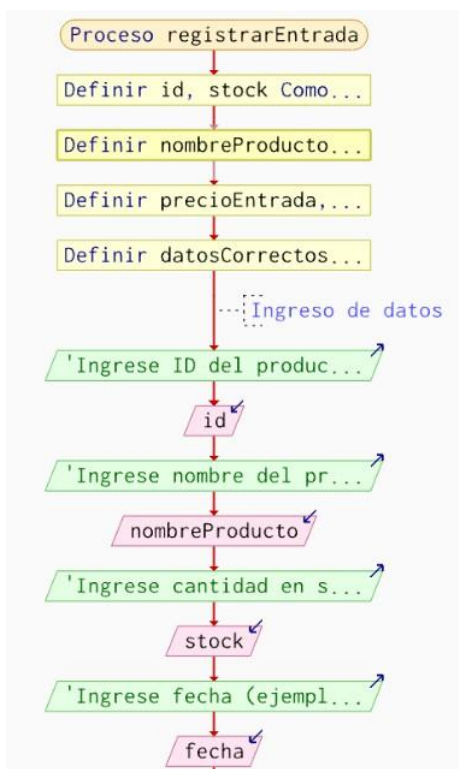
N: Número de nodos

Prueba caja blanca de registro de productos

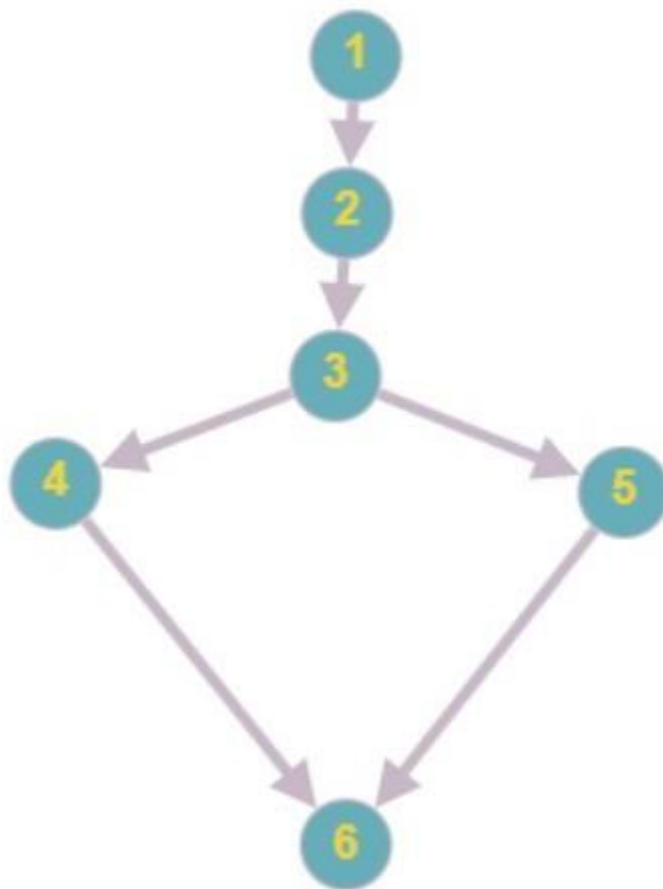
1. CÓDIGO FUENTE

```
void agregarEntrada(){
    Calendar cal;
    int d,m,a;
    cal=dateFecha.getCalendar();
    d=cal.get(Calendar.DAY_OF_MONTH);
    m=cal.get(Calendar.MONTH);
    a=cal.get(Calendar.YEAR)-1900;
    e.setIdproducto(Integer.parseInt(txtidProducto.getText()));
    e.setStock(Integer.parseInt(txtstock.getText()));
    e.setFecha(new Date(a,m,d));
    e.setIdproveedor(Integer.parseInt(txtidproveedor.getText()));
    e.setPrecioE(Double.parseDouble(txtprecioE.getText()));
    e.setPrecioV(Double.parseDouble(txtprecioV.getText()));
    e.setTotal(Double.parseDouble(txtTotal.getText()));
    p.setIdproducto(Integer.parseInt(txtidProducto.getText()));
    p.setPrecioV(Double.parseDouble(txtprecioV.getText()));
    if(dao.insertar(e)&&daoPr.sumarStock(Integer.parseInt(txtidProducto.getText()), Integer.parseInt(txtstock.getText()))){
        MenuPrincipal menu=new MenuPrincipal();
        menu.exito("Entrada Registrada Con Exito");
    }else{
        MenuPrincipal menu=new MenuPrincipal();
        menu.error("No se pudo registrar la entrada");
    }
}
```

2. DIAGRAMA DE FLUJO (DF)



3. GRAFO DE FLUJO (GF)



4. IDENTIFICACIÓN DE LAS RUTAS (Camino básico)

RUTAS

R1: 1 , 2 , 3 , 5 , 6

R2: 1 , 2 , 3 , 4 , 6

5. COMPLEJIDAD CICLOMÁTICA

Se puede calcular de las siguientes formas:

- $V(G) = \text{número de nodos predcados(decisiones)} + 1$
 $V(G) = 1 + 1 = 2$
- $V(G) = A - N + 2$
 $V(G) = 6 - 6 + 2 = 2$

DONDE:

P: Número de nodos predicado

A: Número de aristas

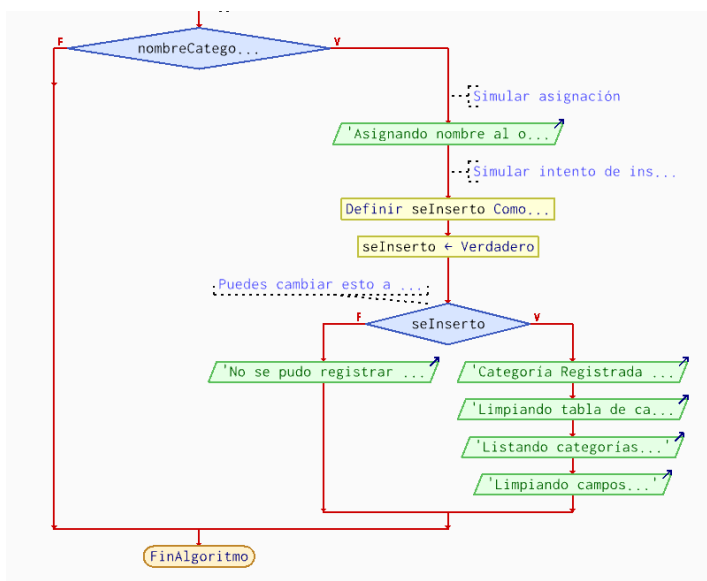
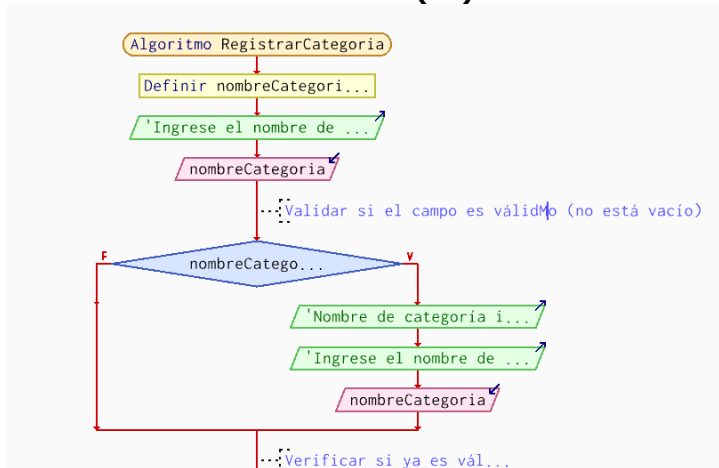
N: Número de nodos

Prueba caja blanca de registro de categorías

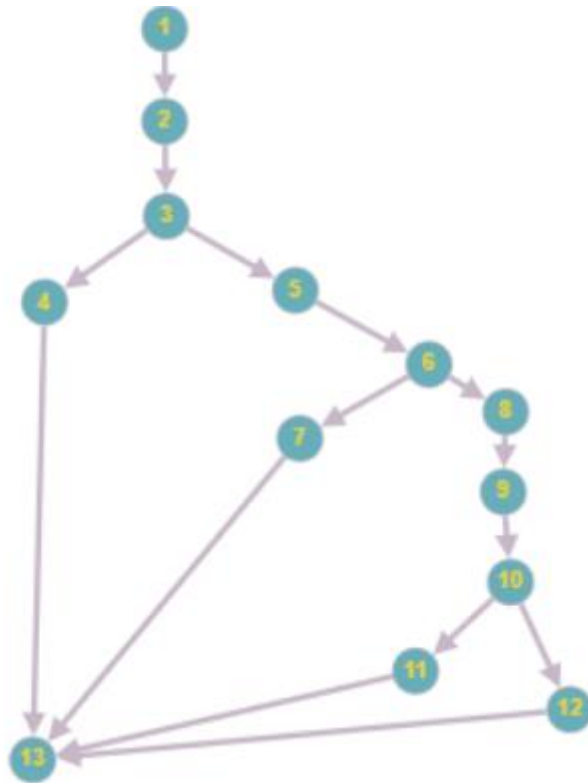
1. CÓDIGO FUENTE

```
private void btnGuardarActionPerformed(java.awt.event.ActionEvent evt) {  
    String nombreCategoria = txtnomCategoria.getText().trim();  
  
    // Validar que solo contenga letras y espacios (opcional)  
    if (!nombreCategoria.matches("[a-zA-ZáéíóúÁÉÍÓÚñÑ\\s]+$")) {...4 lines }  
  
    ct.setNomCategoria(nombreCategoria);  
    if(daoCt.insertar(ct)){  
        m.exito("Categoria Registrada Con Exito");  
    }else{  
        m.error("No se pudo registrar la Categoria");  
    }  
    limpiarTablaCategoria();  
    listarCategorias();  
    limpiarCampos();  
}
```

2. DIAGRAMA DE FLUJO (DF)



3. GRAFO DE FLUJO (GF)



4. IDENTIFICACIÓN DE LAS RUTAS (Camino básico)

RUTAS

R1: 1 , 2 , 3 , 5 , 6 , 8 , 9 , 10 , 12 , 13

R2: 1 , 2 , 3 , 5 , 6 , 8 , 9 , 10 , 11 , 13

R3: 1 , 2 , 3 , 5 , 6 , 7 , 13

R4: 1 , 2 , 3 , 4 , 13

5. COMPLEJIDAD CICLOMÁTICA

Se puede calcular de las siguientes formas:

- $V(G) = \text{número de nodos predcados(decisiones)} + 1$
 $V(G) = 3 + 1 = 4$
- $V(G) = A - N + 2$
 $V(G) = 15 - 13 + 2 = 4$

DONDE:

P: Número de nodos predcado

A: Número de aristas

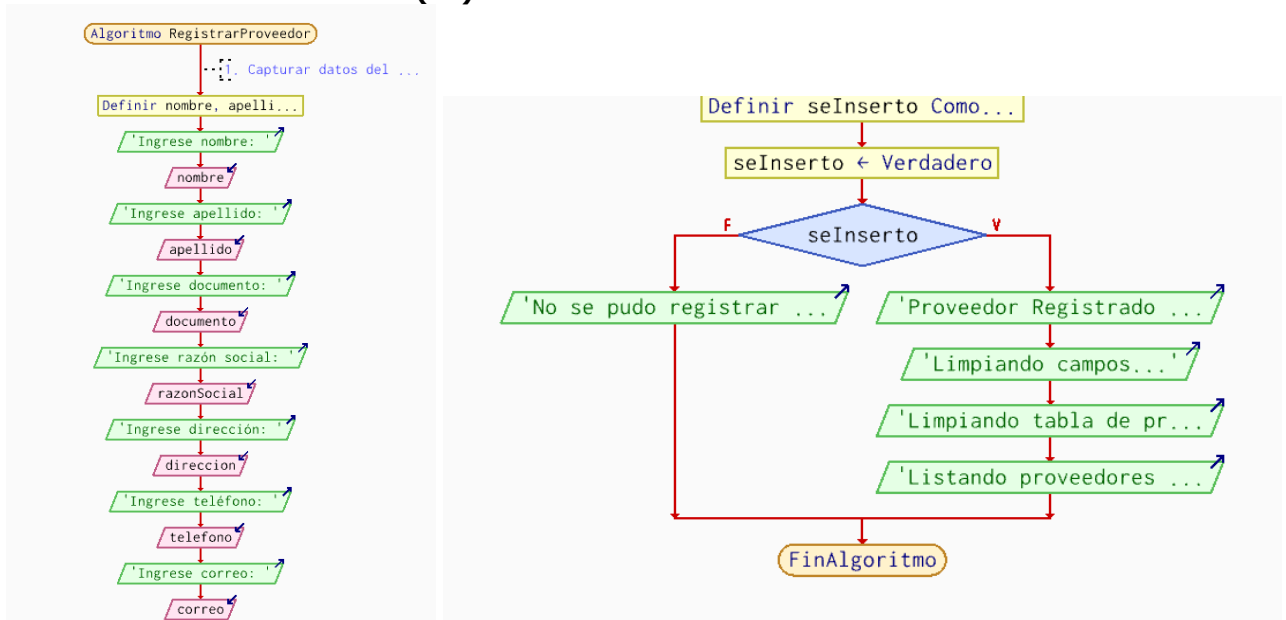
N: Número de nodos

Prueba caja blanca de registro de proveedores

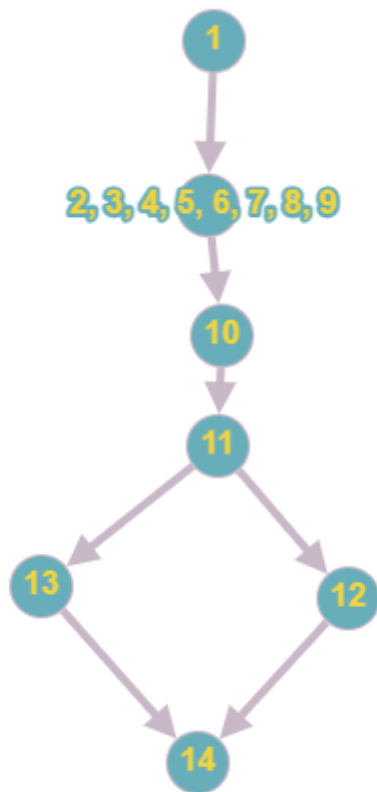
1. CÓDIGO FUENTE

```
private void btnGuardarActionPerformed(java.awt.event.ActionEvent evt) {  
    String nombreCategoria = txtnomCategoria.getText().trim();  
  
    // Validar que solo contenga letras y espacios (opcional)  
    if (!nombreCategoria.matches("[a-zA-ZáéíóúÁÉÍÓÚñÑ\\s]+$")) { ...4 lines }  
  
    ct.setNomCategoria(nombreCategoria);  
    if (daoCt.insertar(ct)) {  
        m.exito("Categoria Registrada Con Exito");  
    } else {  
        m.error("No se pudo registrar la Categoria");  
    }  
    limpiarTablaCategoria();  
    listarCategorias();  
    limpiarCampos();  
}
```

2. DIAGRAMA DE FLUJO (DF)



3. GRAFO DE FLUJO (GF)



4. IDENTIFICACIÓN DE LAS RUTAS (Camino basico)

RUTAS

R1: 1 , 2 , 3 , 5 , 6 , 7 , 8 , 9 , 10 , 11 , 12 , 14

R2: 1 , 2 , 3 , 5 , 6 , 7 , 8 , 9 , 10 , 11 , 13 , 14

5. COMPLEJIDAD CICLOMÁTICA

Se puede calcular de las siguientes formas:

- $V(G) = \text{número de nodos predcados(decisiones)} + 1$
 $V(G) = 1 + 1 = 2$
- $V(G) = A - N + 2$
 $V(G) = 7 - 7 + 2 = 2$

DONDE:

P: Número de nodos predcado

A: Número de aristas

N: Número de nodos