



Universidad de Guadalajara.

Centro Universitario de Ciencias Exactas e Ingenierías.

DIVISIÓN DE TECNOLOGÍAS PARA LA INTEGRACIÓN CIBER HUMANA.

DEPARTAMENTO DE CIENCIAS COMPUTACIONALES.

Practica6



NOMBRE DE LOS ESTUDIANTES:

Padilla Perez Jorge Daray

Juan Jesús Sámano Juárez

Ernesto Macias Flores

NOMBRE DE LA MATERIA: Seminario Sistemas Operativos

SECCION: D04

CICLO ESCOLAR: 2024-A

NOMBRE DEL PROFESOR: Julio Esteban

Codigo:

```
package programa7;

import java.awt.event.ActionEvent;
import java.util.ArrayList;
import java.util.TimerTask;
import javax.swing.Timer;
import javax.swing.JOptionPane;
import javax.swing.JTabbedPane;
import javax.swing.table.DefaultTableModel;
import java.util.Random;
import java.awt.event.KeyEvent;
import java.awt.event.KeyListener;
import javax.swing.JTextField;

public class MainProgram extends javax.swing.JFrame implements KeyListener {

    private Timer aTimer;
    private int count=0;//contador local
    private int countGlobal=0;//contador local
    private int tiempo=0;
    private int tempas=0;
    private int it=0;
    private int iterator_tabla=0;
    private int queue=0;
    private DefaultTableModel model = new DefaultTableModel();
    private DefaultTableModel model2 = new DefaultTableModel();
    private DefaultTableModel model3 = new DefaultTableModel();
    private DefaultTableModel model4 = new DefaultTableModel();
    private DefaultTableModel model5 = new DefaultTableModel();
    private DefaultTableModel model6 = new DefaultTableModel();
    ArrayList<Proceso> procesos;
    Proceso p;

    public MainProgram() {

        procesos = new ArrayList<>();

        initComponents();

        aTimer = new Timer(1000, (ActionEvent e) -> {
            iniciarCronometro();
        });

        addKeyListener(this);
        setFocusable(true);

        this.setVisible(true);
        jTabbedPane1.setEnabledAt(index: jTabbedPane1.indexOfComponent(component: panelIngresar), enabled:true);
        jTabbedPane1.setEnabledAt(index: jTabbedPane1.indexOfComponent(component: panelVisualizar), enabled:true);
        jTabbedPane1.setEnabledAt(index: jTabbedPane1.indexOfComponent(component: panelFinal), enabled:true);
    }

    /**
     * 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")
    // Generated Code

    private void txtInstruccionesActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }

    private void btnTablaTiempoActionPerformed(java.awt.event.ActionEvent evt) {
        jTabbedPane1.setSelectedIndex(index: 2);
        //DECLARAR LA TABLA 4
        while (tblTiempoResultados.getRowCount() > 0) {
            model4.removeRow(row: 0);
        }
        for (int i=0; i<procesos.size(); i++) {
            System.out.println("LEYENDO LA TABLA");
            int n = Integer.parseInt(String.valueOf(tblResultados.getModel().getValueAt(columnIndex: 1, rowIndex: 0)));
            ob[12] = procesos.get(index: 3).resultado;
            ob[13] = procesos.get(index: 3).ilegales;
            ob[14] = procesos.get(index: 3).finalizacion;
            ob[15] = procesos.get(index: 3).retorno;
            ob[16] = procesos.get(index: 3).tiempo;
            ob[17] = procesos.get(index: 3).tempas;
            ob[18] = procesos.get(index: 3).servicio;
            ob[19] = "FINALIZADO";
            model4.addRow(row: new Object[] {
                tblTiempoResultados.setModel(model4);
            });
            tblTiempoResultados.setModel(model4);
        }
    }

    private void txtInstruccionesActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }

    private void txtTiempoActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }

    private void txtNuevosProActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }

    private void btnComenzarActionPerformed(java.awt.event.ActionEvent evt) {
        System.out.println("num procesos: " + procesos.size());

        //DECLARAR LA TABLA 1
        model1.addColumn(columnName: "ID");
        model1.addColumn(columnName: "Tiempo");
        model1.addColumn(columnName: "T Trans");

        //DECLARAR LA TABLA 2
        model2.addColumn(columnName: "ID");
    }
}
```

```

//DECLARAR LA TABLA 2
model2.addColumn(columnName: "ID");
model2.addColumn(columnName: "Operación");
model2.addColumn(columnName: "Resultado");

//DECLARAR LA TABLA 3
model3.addColumn(columnName: "ID");
model3.addColumn(columnName: "Tiempo Bloqueado");
tblProcBloq.setModel(dataModel: model3);

//DECLARAR LA TABLA 4
model4.addColumn(columnName: "ID");
model4.addColumn(columnName: "Operación");
model4.addColumn(columnName: "Resultado");
model4.addColumn(columnName: "T. Llegada");
model4.addColumn(columnName: "T. Finalización");
model4.addColumn(columnName: "T. Retorno");
model4.addColumn(columnName: "T. Respuesta");
model4.addColumn(columnName: "T. Espera");
model4.addColumn(columnName: "T. Servicio");
model4.addColumn(columnName: "ESTADO");
tblTiempoResultados.setModel(dataModel: model4);

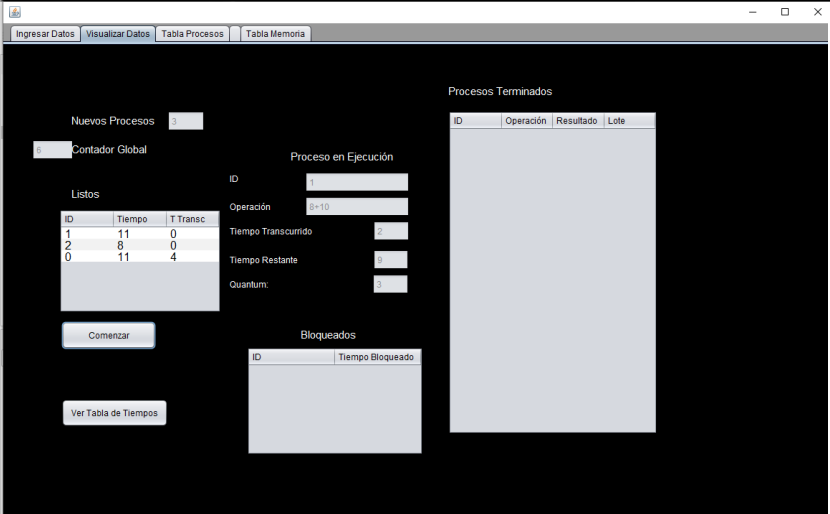
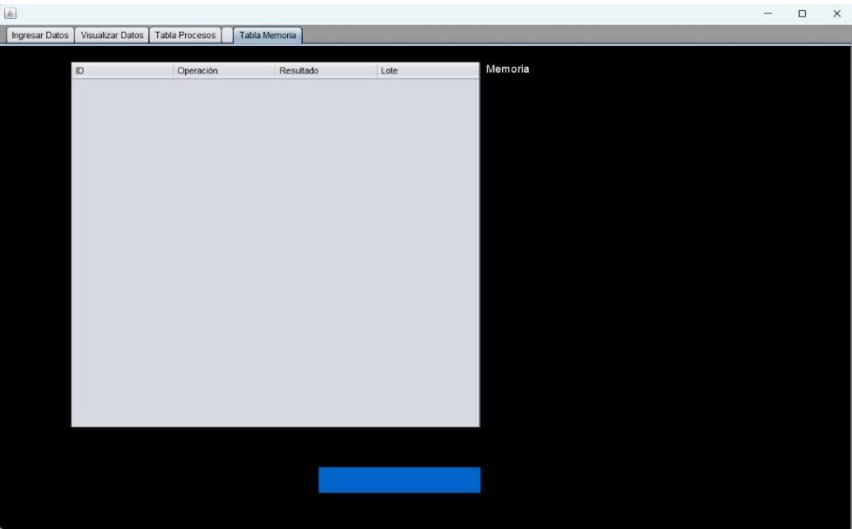
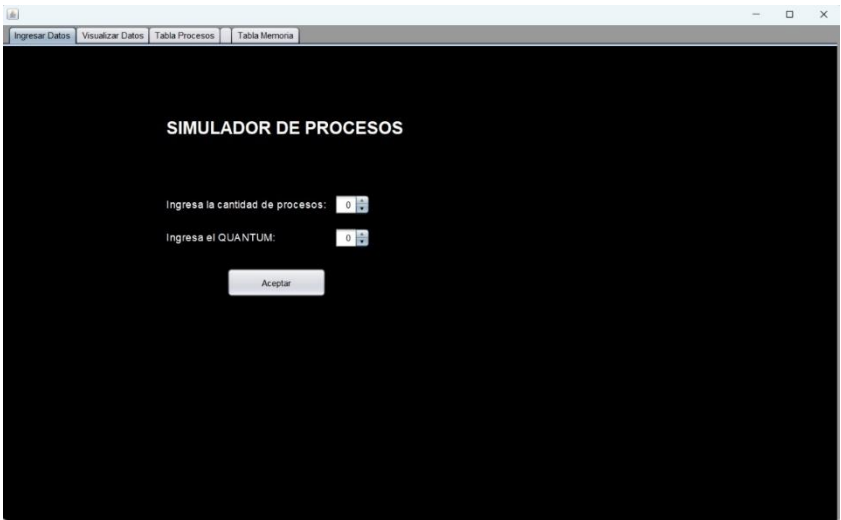
//DECLARAR LA TABLA 5
model5.addColumn(columnName: "No.Marco");
model5.addColumn(columnName: "Espacio");
model5.addColumn(columnName: "Estado");
model5.addColumn(columnName: "Proceso");

int j=0;
Object[] ob = new Object[4];
ob[0]= "0";
ob[1]= "0/5";
ob[2]="S.O.";
ob[3]="...";
model5.addRow(rowData: ob);
ob[0]= "1";
ob[1]= "0/5";
ob[2]="S.O.";
ob[3]="...";
model5.addRow(rowData: ob);

for(int i=0; i<34;i++){
    if(j<procesos.size()){
        int aux1 =(procesos.get(index: j).getTam());
        aux1 = (aux1/5)+1;
        int aux2= procesos.get(index: j).getTam();
        while(aux1!=0){
            if((aux2-5)< 0){
                ob[0]= 1;
                ob[1]= (aux2)+"/5";
                ob[2]= "listo";
                ob[3]= j;
                model5.addRow(rowData: ob);
                aux1--;
                j++;
                //1++
            }
            else{
                ob[0]= 1;
                ob[1]= "5/5";
                ob[2]= "listo";
                ob[3]= j;
                model5.addRow(rowData: ob);
                aux1--;
                1++;
                aux2=aux2-5;
                if(aux2==0){
                    j++;
                }
            }
        }
    }
}

```

Capturas



Juan Jesús Sámano Juárez:

Al realizar este proyecto fue un poco complicado el manejo de frames o también conocidos como marcos, después de haber solucionado este paso fue la elaboración de icono, los cuales en esta actividad son (gallina, zorro, semillas y bote) esto fue relativamente fácil de este punto la siguiente tarea era asignarles un evento de movimiento y arrastre para que el usuario pueda manejarlos y por último agregar las condicionales del juego.

Padilla Perez Jorge Daray:

Para concluir con esta actividad interesante ya que es raro tener que simular que una computadora actual tarde tanto en hacer ciertos procesos y más cuando es algo tan sencillo, el tema de tener varias ventanas fue todo un reto que el compañero Ernesto soluciono, la comunicación en equipo fue importante para la realización de esta actividad, aprendí juntos a mis compañeros y espero que podamos seguir así de comunicados.

Ernesto Macias Flores:

Durante la ejecución de este proyecto, nos encontramos con algunos desafíos al gestionar los marcos o frames. Una vez resueltos, nos enfocamos en el diseño de los íconos necesarios para la actividad: una gallina, un zorro, semillas y un bote. Este proceso resultó relativamente sencillo. A continuación, nos centramos en dotar a cada ícono con eventos de movimiento y capacidad de arrastre, brindando así al usuario el control necesario. Finalmente, completamos el proyecto integrando las condiciones del juego para una experiencia de usuario completa.