Imagen que contiene Logotipo

Descripción generada automáticamenteTexto

Descripción generada automáticamente con confianza media

Benemérita Universidad Autónoma de Puebla

“Pensar bien, para vivir mejor”

Facultad Ciencias de la Computación

Programación Concurrente y Paralela

Práctica 3

* Profesor: Cerón Garnica Carmen

• Equipo

Rojas Luna Alexa Michelle 202074633

Ponce Lara José Abel 202051988

Urbano Nieva Raúl 202059928

Antonio Ambrosio Jesús Noé 202062848

**Objetivo de la práctica**

Utilizar semáforos generales y binarios.

**Introducción**

Para implementar un **semáforo** **binario** yun **semáforo general** se usan las primitivas propias de java. Con esto quedará demostrado que las primitivas en java tienen como mínimo el mismo poder de expresividad que los semáforos y que cualquier problema que podamos resolver con semáforos también podremos resolverlos con java.

La ventaja de java con respecto a los semáforos es que sus primitivas son más fáciles de utilizar y no están dispersas por el código de los objetos cliente, sino centradas en los objetos que juegan el papel de servidores con lo que el código se hace más fácilmente modificable. No obstante, si uno se siente más cómodo podría utilizar el paquete semáforo que se implementa a continuación para sincronizar sus programas en java. Utilizamos nombres en mayúsculas para wait() y signal() para evitar confusiones con el wait() propio de java.

**Desarrollo de la práctica**

1. Realizar los grafos del cuestionario 3 implementando

semáforos.

1. Realizar el grafo de ejemplo modo gráfico.
2. Implementar en modo gráfico con semáforos dos grafos del cuestionario tres.
3. Realizar productor/consumidor con semáforos en java modo gráfico.
4. Realizar los ejercicios del examen uno con semáforos.

**Código**

import java.awt.Graphics;

/\*\*

\*

\* @author crish

\*/

public class A extends JFrameEquipo implements Runnable{

@Override

public void run(){

try{

mutex.WAIT();

aviso[i] = "Soy el proceso #1 y la instruccion A";

orden[i] = "A";

i++;

mutex.SIGNAL();

uno.SIGNAL();

dos.SIGNAL();

tres.SIGNAL();

}catch(Exception ex){}

}

}

import java.awt.Graphics;

/\*\*

\*

\* @author crish

\*/

public class B extends JFrameEquipo implements Runnable{

public void run(){

try{

uno.WAIT();

mutex.WAIT();

aviso[i] = "Soy el proceso #2 y la instruccion B";

orden[i] = "B";

i++;

mutex.SIGNAL();

cuatro.SIGNAL();

cinco.SIGNAL();

}catch(Exception e){}

}

}

import java.awt.Graphics;

/\*\*

\*

\* @author crish

\*/

public class C extends JFrameEquipo implements Runnable {

public void run(){

try{

dos.WAIT();

cuatro.WAIT();

siete.WAIT();

mutex.WAIT();

aviso[i] = "Soy el proceso #3 y la instruccion C";

orden[i] = "C";

i++;

mutex.SIGNAL();

seis.SIGNAL();

}catch(Exception e){}

}

}

import java.awt.Graphics;

/\*\*

\*

\* @author crish

\*/

public class D extends JFrameEquipo implements Runnable {

public void run() {

try{

tres.WAIT();

mutex.WAIT();

aviso[i] = "Soy el proceso #4 y la instruccion D";

orden[i] = "D";

i++;

mutex.SIGNAL();

siete.SIGNAL();

ocho.SIGNAL();

}catch(Exception e){}

}

}

import java.awt.Graphics;

/\*\*

\*

\* @author crish

\*/

public class E extends JFrameEquipo implements Runnable{

public void run(){

try{

cinco.WAIT();

seis.WAIT();

ocho.WAIT();

mutex.WAIT();

aviso[i] = "Soy el proceso #5 y la instruccion E";

orden[i] = "E";

i++;

mutex.SIGNAL();

nueve.SIGNAL();

diez.SIGNAL();

//Thread.sleep(250);

}catch(Exception e){}

}

}

import java.awt.Graphics;

/\*\*

\*

\* @author crish

\*/

public class F extends JFrameEquipo implements Runnable{

public void run(){

try{

nueve.WAIT();

mutex.WAIT();

aviso[i] = "Soy el proceso #6 y la instruccion F";

orden[i] = "F";

i++;

mutex.SIGNAL();

once.SIGNAL();

}catch(Exception e){}

}

}

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

/\*\*

\*

\* @author crish

\*/

public class G extends JFrameEquipo implements Runnable {

public void run(){

try{

diez.WAIT();

once.WAIT();

mutex.WAIT();

aviso[i] = "Soy el proceso #7 y la instruccion G";

orden[i] = "G";

mutex.SIGNAL();

}catch(Exception ex){}

}

}

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

import java.awt.Dimension;

import java.awt.Graphics;

import javax.swing.JFrame;

import static javax.swing.WindowConstants.EXIT\_ON\_CLOSE;

/\*\*

\*

\* @author crish

\*/

public class JFrameEquipo extends javax.swing.JFrame {

protected static final Semaforo uno = new Semaforo(0);

protected static final Semaforo dos = new Semaforo(0);

protected static final Semaforo tres = new Semaforo(0);

protected static final Semaforo cuatro = new Semaforo(0);

protected static final Semaforo cinco = new Semaforo(0);

protected static final Semaforo seis = new Semaforo(0);

protected static final Semaforo siete = new Semaforo(0);

protected static final Semaforo ocho = new Semaforo(0);

protected static final Semaforo nueve = new Semaforo(0);

protected static final Semaforo diez = new Semaforo(0);

protected static final Semaforo once = new Semaforo(0);

protected static final Semaforo mutex = new Semaforo(1);

public static int i = 0;

public static String [] aviso = new String[7];

public static String [] orden = new String[7];

/\*\*

\* Creates new form JFrameEquipo

\*/

public JFrameEquipo() {

initComponents();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is alway \* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jScrollPane1 = new javax.swing.JScrollPane();

jTextArea1 = new javax.swing.JTextArea();

jlblA = new javax.swing.JLabel();

jlblB = new javax.swing.JLabel();

jlblE = new javax.swing.JLabel();

jlblC = new javax.swing.JLabel();

jlblD = new javax.swing.JLabel();

jlblG = new javax.swing.JLabel();

jlblF = new javax.swing.JLabel();

jLabel1 = new javax.swing.JLabel();

jButton1 = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

setTitle("unoEquipo");

setMinimumSize(new java.awt.Dimension(850, 720));

setPreferredSize(new java.awt.Dimension(0, 0));

setResizable(false);

setSize(new java.awt.Dimension(836, 500));

getContentPane().setLayout(null);

jTextArea1.setEditable(false);

jTextArea1.setColumns(20);

jTextArea1.setRows(5);

jScrollPane1.setViewportView(jTextArea1);

getContentPane().add(jScrollPane1);

jScrollPane1.setBounds(12, 13, 814, 129);

jlblA.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);

getContentPane().add(jlblA);

jlblA.setBounds(90, 380, 40, 20);

jlblB.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);

getContentPane().add(jlblB);

jlblB.setBounds(290, 210, 30, 20);

jlblE.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);

getContentPane().add(jlblE);

jlblE.setBounds(530, 380, 30, 20);

jlblC.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);

getContentPane().add(jlblC);

jlblC.setBounds(290, 380, 40, 30);

jlblD.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);

getContentPane().add(jlblD);

jlblD.setBounds(290, 560, 40, 20);

jlblG.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);

getContentPane().add(jlblG);

jlblG.setBounds(720, 560, 40, 20);

jlblF.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);

getContentPane().add(jlblF);

jlblF.setBounds(730, 210, 37, 16);

jLabel1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/colab3\_graficografo/GRAFF.png"))); // NOI18N

jLabel1.setMinimumSize(new java.awt.Dimension(814, 814));

getContentPane().add(jLabel1);

jLabel1.setBounds(10, 160, 820, 460);

jButton1.setText("Iniciar Hilos");

jButton1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton1ActionPerformed(evt);

}

});

getContentPane().add(jButton1);

jButton1.setBounds(710, 640, 100, 25);

pack();

setLocationRelativeTo(null);

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

try{

jTextArea1.setText("");

Thread A = new Thread(new A());

Thread B = new Thread(new B());

Thread C = new Thread(new C());

Thread D = new Thread(new D());

Thread E = new Thread(new E());

Thread F = new Thread(new F());

Thread G = new Thread(new G());

A.start();

B.start();

C.start();

D.start();

E.start();

F.start();

G.start();

Thread.sleep(1000);

jTextArea1.append(aviso[0]+"\n");

jTextArea1.append(aviso[1]+"\n");

jTextArea1.append(aviso[2]+"\n");

jTextArea1.append(aviso[3]+"\n");

jTextArea1.append(aviso[4]+"\n");

jTextArea1.append(aviso[5]+"\n");

jTextArea1.append(aviso[6]+"\n");

jlblA.setText(orden[0]);

jlblB.setText(orden[1]);

jlblC.setText(orden[3]);

jlblD.setText(orden[2]);

jlblE.setText(orden[4]);

jlblF.setText(orden[5]);

jlblG.setText(orden[6]);

}catch(InterruptedException ex){}

i = 0;

}

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(JFrameEquipo.class.getName()).log(java.util.logging.Level.SEVERE, null,

ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(JFrameEquipo.class.getName()).log(java.util.logging.Level.SEVERE, null,

ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(JFrameEquipo.class.getName()).log(java.util.logging.Level.SEVERE, null,

ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(JFrameEquipo.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 JFrameEquipo().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton jButton1;

private javax.swing.JLabel jLabel1;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JTextArea jTextArea1;

private javax.swing.JLabel jlblA;

private javax.swing.JLabel jlblB;

private javax.swing.JLabel jlblC;

private javax.swing.JLabel jlblD;

private javax.swing.JLabel jlblE;

private javax.swing.JLabel jlblF;

private javax.swing.JLabel jlblG;

// End of variables declaration

}

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

/\*\*

\*

\* @author crish

\*/

/\*\*

\*

\* @author Alumno

\*/

public class Semaforo {

private int contador;

public Semaforo(int n)

{

this.contador = n;

}

public synchronized void WAIT()

{

while(contador == 0)

{

try

{

wait();

}

catch (InterruptedException e) {}

}

contador--;

}

public synchronized void SIGNAL()

{

contador=1;

notify();

}

}

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Main.java to edit this template

\*/

package colab3\_graficografo;

/\*\*

\*

\* @author crish

\*/

public class Colab3\_GraficoGrafo {

/\*\*

\* @param args the command line arguments

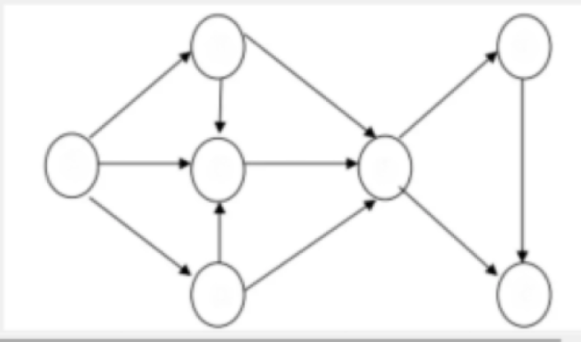
\*/

public static void main(String[] args) {

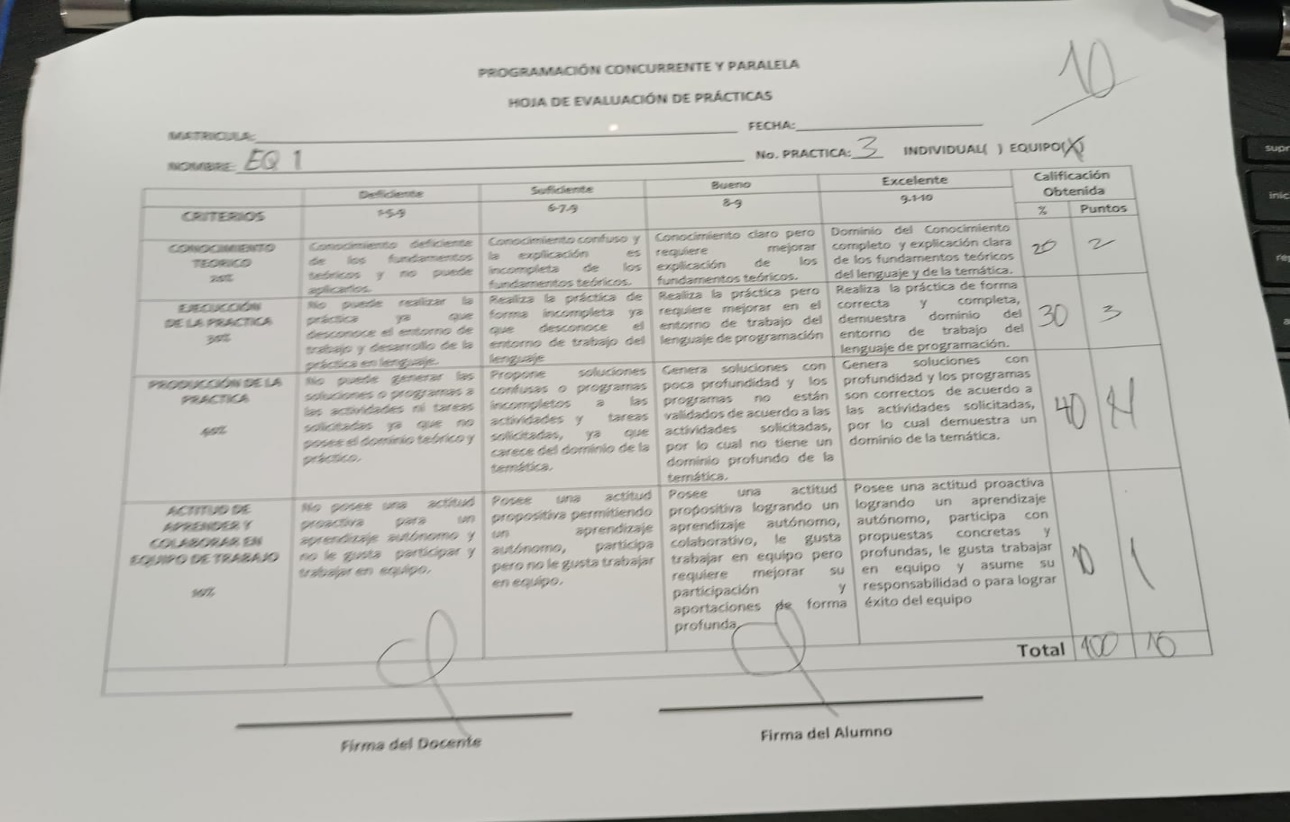
// TODO code application logic here

}

}



**Hoja firmada**

****