



Universidad Simón Bolívar

Redes de Computadoras I - CI-4835

Daniel Robayo 18-11086

Gabriel Chaurio 17-10126

Sinai Guzmán 16-10505

## **Laboratorio #1: Máquinas de Estados y Protocolos de Comunicación**

### **INTRODUCCIÓN.**

Una máquina de estados es un modelo matemático que se utiliza para describir el comportamiento de un sistema. En dicha máquina, el sistema se modela como una serie de estados, transiciones y eventos. Los estados representan las diferentes condiciones o modos posibles en los que el sistema puede encontrarse, mientras que las transiciones representan los cambios de estado.

Los protocolos de comunicación son conjuntos de reglas que especifican cómo los nodos de una red se comunican entre sí. Los protocolos definen el formato de los mensajes que se intercambian, así como las reglas para el envío y su recepción.

Un ejercicio práctico para la construcción de una máquina de estado es realizar una analogía entre estas y una red. Esto debido a que si tenemos varias máquinas de estado y algunas de sus transiciones dependen del estado de varias máquinas, se deben realizar una comunicación para que se llegue al resultado de la mejor manera.

## OBJETIVOS.

- Modelar dos autómatas independientes que en conjunto resuelvan un problema.
- Realizar un programa en lenguaje C en donde se modelen los autómatas.
- Profundizar en el entendimiento de los protocolos de comunicación, formando una analogía entre máquinas de estados y comunicación entre procesos.

## DESARROLLO.

Se requiere la definición de dos autómatas capaces de resolver el siguiente problema:

“Se tienen dos jarras de agua, una tiene una capacidad de 5 litros y otra de 3 litros (esto quiere decir que NO cabe en ellas ni una gota más de agua). ¿Cómo podemos conseguir que en uno de los dos recipientes haya 4 litros de agua exactos?”

Para la solución, se definen las siguientes operaciones y autómatas.

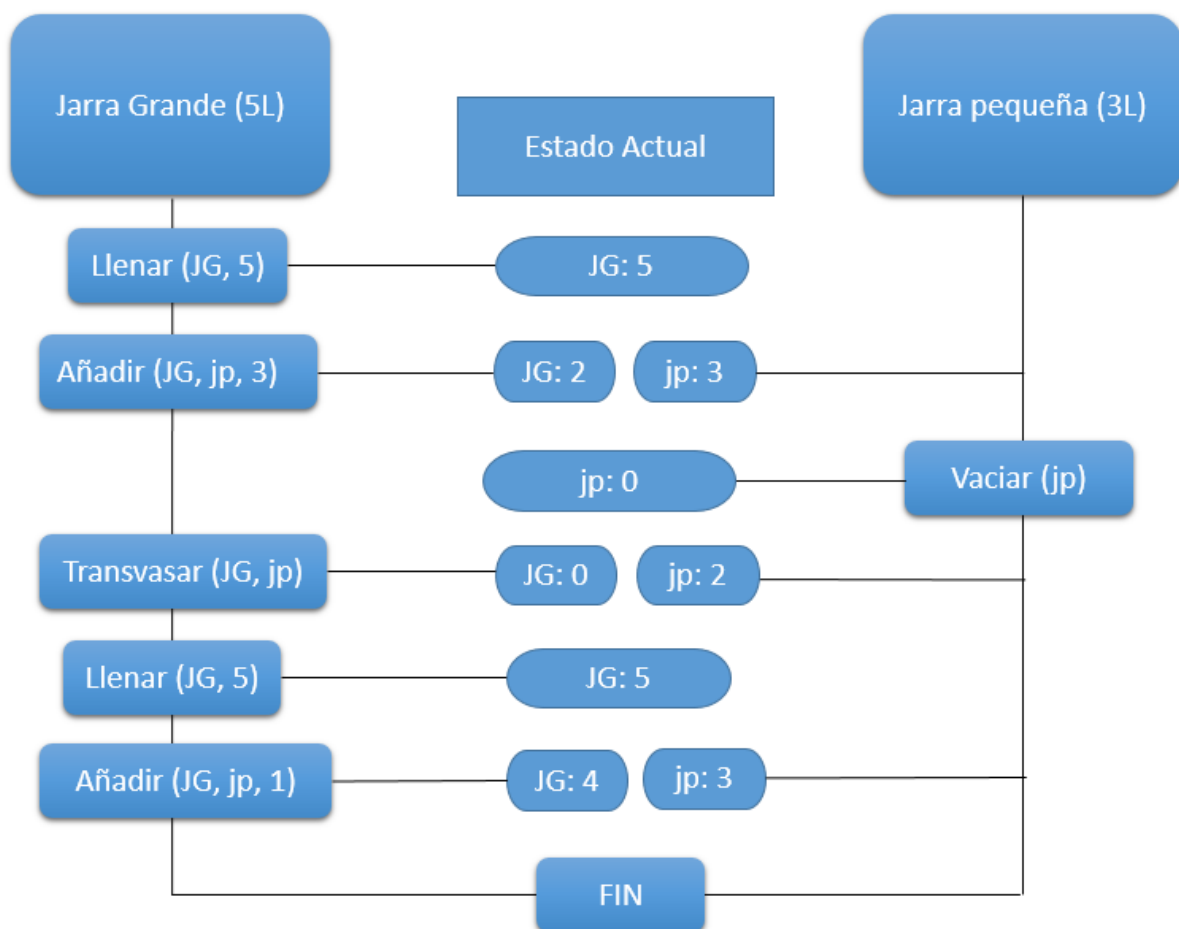
Son dos autómatas JG y jp, donde sus estados representan la cantidad de litros de la jarra grande y la jarra pequeña respectivamente. Para ambos autómatas se definen las siguientes transiciones.

- Llenar (JARRA,m): llenar JARRA con m litros de agua
- Añadir (JARRA\_i, JARRA\_j, m): añadir m litros de JARRA\_i a JARRA\_j
- Transvasar (JARRA\_i,JARRA\_j): pasar el agua de JARRA\_i a JARRA\_j
- Vaciar (JARRA): vaciar JARRA

Las operaciones que involucran dos jarras modifican el estado de una dada la acción de otra, por lo que para que la operación se realice correctamente la jarra destino debe “avisarle” a la jarra origen que está listo para recibir. Se presenta la siguiente solución.

1. Llenar(JG, 5)
2. Añadir (JG, jp, 3)
3. Vaciar (jp)
4. Transvasar (JG,jp)
5. Llenar(JG, 5)
6. Añadir (JG, jp, 1)

El diagrama de estado de cada autómatas en donde se realiza cada operación se muestra a continuación.



Además de la definición de los autómatas, se realizó un programa escrito en C en donde se implementaron los autómatas descritos. El programa realiza exactamente las operaciones y los cambios de estados de cada autómata y funciona de la siguiente manera:

1. El proceso padre, el cual representa a JG, crea un proceso hijo el cual representa a jp
2. Cada proceso realiza sus acciones individuales de manera independiente. En el momento de realizar una operación en conjunto, se envía un mensaje entre los procesos usando dos pipes. El primer pipe crea una conexión entre JG y jp y el otro entre jp y JG.
3. De esta manera, cada vez que un proceso está esperando un mensaje, este se encuentra en *busy wait* mientras está leyendo el pipe.

El programa corre estrictamente en un sistema operativo linux, no recibe valores de entrada y muestra en la consola las operaciones y los estados de los autómatas. Para compilar se debe usar el comando `$gcc protocolo.c -o "nombre"`. Por último para ejecutar el programa se debe usar el comando `$/ "nombre"`

## **CONCLUSIONES.**

En esta actividad aprendimos a modelar un problema, que a primera vista su solución es algorítmica, usando máquinas de estados. Se ideó una manera de representar dichas máquinas utilizando procesos y se apreció el funcionamiento mediante la analogía aplicada al utilizar procesos y pipes. Además, aprendimos cómo funciona, a alto nivel y simplificando gran cantidad de conceptos, como son las operaciones en una red de computadoras.

Por último, se apreció como operaciones que son complejas como la creación de procesos y la comunicación entre estos permiten la sincronización y comunicación de nodos en una red.