



**Universidad Nacional  
Autónoma de México**

**Facultad de Ciencias**



**Licenciatura en Ciencias de la Computación**

**Computación Distribuida**

## **Práctica 1**

**(Reporte)**

### **Equipo:**

Elizalde Maza Jesús Eduardo 321031686

Peredo López Citlalli Abigail 321161022

Fernando Michel Tavera  
Luis Mario Escobar Rosales  
Brenda Ayala Flores  
David Ortega Medina

10 de Septiembre del 2025

El principal objetivo de la práctica es modelar algoritmos distribuidos utilizando SimPy. Para esto se desarrollaron los canales de comunicación que permiten que los nodos intercambien mensajes, y distintos tipos de nodos distribuidos ya que cada tipo de nodo ejecuta un algoritmo distinto.

Los algoritmos que se solicitaron realizar son:

- Código vecinos para proceso  $p_i$   
Este algoritmo lo desarrollamos en la clase `NodoVecinos`, el cual tiene como objetivo que cada nodo descubra no solo a sus vecinos directos, sino también a los vecinos de segundo nivel. Cada nodo inicia enviando a sus vecinos directos la lista de sus propios vecinos, al recibir un mensaje, el nodo lo interpreta como la lista de vecinos de un vecino directo y esta información se guarda en el conjunto `identifiers` y de esta manera logra descubrir quienes son los vecinos de sus vecinos.
- Código vecinos topología para  $p_i$   
Este algoritmo se desarrolló en la clase `NodoTopologia`, el cual a partir del intercambio de mensajes, cada nodo reconstruye la red completa. Para ello cada nodo envía al inicio el par formado por su identificador y la lista de sus vecinos directos. Cuando reciben información de un nodo desconocido lo agrega a su conjunto de procesos conocidos y actualiza su lista de canales conocidos con las nuevas aristas. Posteriormente reenvía esa información a sus vecinos exceptuando al que se la envió originalmente para evitar ecos innecesarios. El algoritmo incluye una condición de paro que se cumple cuando el nodo verifica que todos los procesos y canales registrados forman parte de la red, en cuyo caso deja de ejecutar el proceso.
- Broadcast ingenuo para el proceso  $p_i$   
Este algoritmo se desarrolló en la clase `NodoBroadcast`, el cual el nodo distinguido inicia con un mensaje y lo envía a todos los vecinos. Cada nodo que recibe por primera vez el mensaje lo almacena, marca que ya lo ha visto y lo reenvía a sus propios vecinos.