

## **Participantes:**

Keury Ramirez **2023-1101**

Hugo Donator **2023-1082**

Jahaziel Lorenzo **2023-1298**

Reyning Perdomo **2023-1110**

**Carrera:** Desarrollo de software

**Materia:** Programación Paralela

**Docente:** Erick Leonardo Perez Veloz

**Tema:** Elección de Tema Final (1-2)

**Fecha:**

6/4/2025

## **Propuesta 1: Sistema de Gestión de Tráfico Urbano en Tiempo Real**

Este proyecto consiste en crear un sistema que ayude a gestionar el tráfico en una ciudad de forma inteligente. Se utilizarán sensores, cámaras y otros dispositivos para recoger datos sobre el flujo de vehículos en diferentes puntos de la ciudad. La idea es procesar esta información en paralelo para tomar decisiones rápidas y mejorar la circulación. El sistema permitirá ajustar semáforos, informar a los conductores y detectar zonas de congestión en tiempo real.

### **1) Ejecución simultánea de múltiples tareas**

Se procesarán los datos de cada sensor y cámara de forma paralela, lo que permite analizar varias zonas de la ciudad al mismo tiempo.

### **2) Compartición de datos entre tareas**

La información recopilada se compartirá entre distintos módulos, por ejemplo, entre los módulos que controlan los semáforos y los que generan reportes en tiempo real.

### **3) Estrategias de paralelización**

Se podrán probar diferentes formas de paralelizar el proceso, como el uso de hilos (threads) o procesos separados, e incluso el procesamiento distribuido en varios servidores.

### **4) Capacidad de escalar con más recursos**

Si la ciudad crece o se agregan más sensores, el sistema se puede ampliar fácilmente añadiendo más nodos o servidores.

### **5) Métricas de evaluación del rendimiento**

Se evaluarán:

- El tiempo de respuesta.
- La eficiencia en la reducción de congestión.
- La cantidad de datos procesados por segundo.
- El uso de recursos.

### **6) Aplicación en escenarios del mundo real**

Este sistema es muy útil para ciudades modernas, donde mejorar la movilidad urbana es clave para reducir la contaminación y los atascos de tráfico.

## **Propuesta 2: Simulador de Mercado Bursátil Paralelo**

Este proyecto consiste en simular el comportamiento de un mercado bursátil en el que varios agentes realizan operaciones de compra y venta de acciones de manera paralela.

### **1) Ejecución simultánea de múltiples tareas**

El simulador utiliza tareas paralelas (por ejemplo, `Task.Run()`, `Parallel.For`) para permitir que los agentes ejecuten operaciones de compra/venta de acciones simultáneamente. Cada agente es representado por una tarea que se ejecuta de manera independiente, simulando un mercado donde las transacciones ocurren al mismo tiempo.

### **2) Necesidad de compartir datos entre tareas**

Las tareas (agentes) necesitan acceder a los precios compartidos de las acciones en el objeto `Mercado`. Para mantener la consistencia de estos precios, se debe implementar una sincronización de datos, como el uso de `lock` o estructuras de datos concurrentes, asegurando que los agentes no modifiquen los precios simultáneamente de manera inconsistente.

### **3) Permitir la exploración de diferentes estrategias de paralelización**

El proyecto permite comparar diferentes estrategias de paralelización, como la ejecución mediante `Parallel.For`, el uso de `Task.Run` para cada agente, o incluso paralelización mediante técnicas de *divide y vencerás* (como particionar la lista de acciones).

### **4) Tener la capacidad de escalar con más recursos**

El simulador es escalable. Si se aumenta el número de agentes o núcleos de CPU disponibles, el sistema puede ejecutar más tareas en paralelo sin perder rendimiento. Además, el sistema puede escalar horizontalmente, distribuyendo la carga a través de múltiples servidores o contenedores si se implementa en la nube.

### **5) Incluir métricas de evaluación del rendimiento**

Se pueden incluir métricas de rendimiento como:

- Tiempo total de simulación.
  - Operaciones realizadas por segundo.
  - Rendimiento por agente.
  - Eficiencia de la paralelización comparando la simulación secuencial vs paralela.
- Estas métricas ayudan a evaluar el impacto del paralelismo en la eficiencia y a ajustar las estrategias de paralelización para maximizar el rendimiento.

## 6) Aplicación a escenarios del mundo real

El simulador es una representación simplificada del mercado bursátil, donde los agentes compran y venden acciones, lo que se asemeja a escenarios reales de trading. El proyecto puede aplicarse a simuladores financieros, modelado de precios de mercado y evaluación de estrategias de inversión en un contexto profesional o educativo.

