

UNIVERSIDAD AUTÓNOMA "JUAN MISAE SARACHO"

**FACULTAD DE CIENCIAS INTEGRADAS DE
YACUIBA**

**CARRERA DE INGENIERÍA INFORMÁTICA
— 8.º SEMESTRE**

**GUÍA OFICIAL DEL PROYECTO FINAL
INTEGRADOR (ABP – FBC)**

ASIGNATURA: TEL420 - Sistemas Paralelos

DOCENTE: Ing. Elias Cassal Baldiviezo

GUÍA DE INSTRUCCIONES PARA EL ESTUDIANTE: PROYECTO INTEGRADOR DE COMPUTACIÓN PARALELA Y DISTRIBUIDA

1. IDENTIFICACIÓN Y DATOS GENERALES DE LA ACTIVIDAD

- **Modalidad pedagógica:** Aprendizaje Basado en Proyectos (ABP) bajo el enfoque de Formación Basada en Competencias (FBC).
- **Conformación de equipos:** Grupos de máximo **tres (3) estudiantes**, organizados por afinidad.
- **Premisa fundamental:** Innovación tecnológica, reproducibilidad y pertinencia contextual orientada a la resolución de problemas reales del Gran Chaco, el departamento de Tarija y Bolivia.
- **Infraestructura de desarrollo asignada:** Laboratorio de Informática — Equipos **Dell OptiPlex Power Plus 7010** (Procesador Intel Core i7, 32 GB de Memoria RAM, Red Gigabit Ethernet, Sistema Operativo base Ubuntu Linux / WSL2).

2. PROPÓSITO DEL PROYECTO Y COMPETENCIA A DEMOSTRAR

El propósito del proyecto final integrador es diseñar, desplegar, programar y evaluar un **sistema de cómputo paralelo o distribuido virtualizado/contenerizado**, utilizándolo como motor computacional para resolver un problema real y complejo de simulación matemática o procesamiento masivo de datos.

Competencia General Demostrada:

"Resuelve problemas computacionales complejos de alto impacto regional mediante el diseño, despliegue y optimización de infraestructuras paralelas (cústeres virtualizados/contenerizados) y la implementación de algoritmos concurrentes bajo

estándares de memoria compartida, memoria distribuida o aceleración por hardware, evaluando cuantitativamente el rendimiento y la escalabilidad bajo principios de ética y responsabilidad social."

3. ARQUITECTURA TÉCNICA E INFRAESTRUCTURA DEL LABORATORIO

Cada equipo aprovechará la capacidad de cómputo de las estaciones **Dell OptiPlex 7010 (Core i7 con 32 GB de RAM)** para implementar la infraestructura paralela bajo una de las siguientes dos modalidades:

OPCIONES ARQUITECTÓNICAS DE DESPLIEGUE

- ```
+-----+
| OPCIÓN A: CÚSTER VIRTUALIZADO / CONTENERIZADO MULTINODO (Docker / Apptainer)
| * Creación de un clúster de al menos 3 nodos (1 Master + 2 Workers).
| * Configuración de red virtual interna tipo bridge con SSH sin contraseña.
| * Entorno OpenMPI y compiladores GCC/G++ empaquetados en contenedores.
+-----+
| OPCIÓN B: ENTORNO HÍBRIDO HADOOP/SPARK O CLÚSTER SLURM VIRTUALIZADO
| * Nodos virtuales gestionados por KVM/Multipass o Docker Compose.
| * Gestor de colas Slurm o clúster de procesamiento distribuido Apache Spark.
| * Asignación estricta de memoria (mínimo 8 GB por nodo virtual).
+-----+
```

#### Requerimientos Técnicos Mínimos de Software:

1. **Sistema Operativo:** Ubuntu Server 22.04 LTS o superior (nativo, en partición o bajo entorno de virtualización/WSL2).
2. **Cómputo en Memoria Compartida:** Implementación con **OpenMP** en C/C++ (control explícito de directivas, variables de ámbito y reducción).
3. **Cómputo en Memoria Distribuida:** Implementación con **OpenMPI** en C/C++ o **PyMPI** (comunicaciones punto a punto no bloqueantes y comunicaciones colectivas Scatter/Gather/Reduce).
4. **Opcional / Plus de Innovación:** Aceleración por GPU mediante **CUDA/PyCUDA** (para equipos con tarjeta gráfica NVIDIA) o integración con **Apache Spark/Slurm**.

## 4. LÍNEAS TEMÁTICAS SUGERIDAS (CONTEXTO REGIONAL YACUIBA Y GRAN CHACO)

El proyecto debe abandonar los ejercicios abstractos de juguete y responder a un problema de ingeniería concreto. Se plantean las siguientes líneas prioritarias:

1. **Simulación y Monitoreo Ambiental en la Cuenca del Río Pilcomayo:**
  - Modelado bidimensional paralelo de dispersión de sedimentos, metales pesados o contaminantes hídricos mediante diferencias finitas.
2. **Análisis Geoespacial Masivo de Focos de Calor e Incendios en el Chaco:**
  - Procesamiento distribuido de imágenes satelitales multiespectrales (Sentinel-2 / Landsat) para el cálculo de índices NDVI o NBR (Normalized Burn Ratio) a gran escala.
3. **Optimización Logística y Redes de Transporte Transfronterizo:**
  - Algoritmo paralelo de ruteo vehicular (VRP/TSP paralelo) para transporte de granos, agroindustria o hidrocarburos entre Bolivia, Argentina y Paraguay.
4. **Agrometeorología Predictiva y Modelado Climático Local:**
  - Simulación estocástica masiva (Monte Carlo paralelo) de balance hídrico y estrés térmico en cultivos de maíz y soya bajo escenarios de sequía en la llanura chaqueña.
5. **Visión Artificial Distribuida para Sanidad Vegetal:**
  - Procesamiento paralelo y clasificación masiva de imágenes foliares para detección temprana de plagas en la agricultura regional.

## 5. FASES DE DESARROLLO DEL PROYECTO

+-----+

| FASE 1: Propuesta Técnica y Selección de Problema (Semana 1)

| FASE 2: Despliegue de Infraestructura y Contenerización (Semana 2-3)

| FASE 3: Desarrollo Algorítmico Paralelo e Híbrido (Semanas 4 - 5)

| FASE 4: Benchmarking, Pruebas de Carga y Evaluación de Rendimiento (Semana 6)

| FASE 5: Entrega Final de Repositorio y Defensa Técnica (Semana 7 - 8 )

+-----+

### Fase 1: Perfil de Propuesta Técnica

- Definición del problema regional, justificación del volumen computacional requerido y selección del modelo matemático/algorítmico.

- Selección de la arquitectura paralela a utilizar (Memoria compartida, distribuida o híbrida).

## Fase 2: Construcción de la Infraestructura de Cómputo

- Configuración de la red y aprovisionamiento de nodos virtuales en el Dell OptiPlex 7010 aprovechando los 32 GB de RAM.
- Construcción del Dockerfile, archivo de Compose o receta de aprovisionamiento que garantice la replicación del entorno en cualquier máquina.

## Fase 3: Programación y Paralelización

- Desarrollo de la versión **secuencial de referencia** ( $T_1$ ).
- Desarrollo de la versión **paralela optimizada** ( $T_p$ ) empleando OpenMP, MPI o una solución híbrida (OpenMP + MPI).
- Manejo estricto de la sincronización: prevención comprobada de condiciones de carrera (*race conditions*), bloqueos (*deadlocks*) y falsas comparticiones (*false sharing*).

## Fase 4: Experimentación y Benchmarking Científico

- Ejecución de pruebas con variaciones sistemáticas:
- Número de hilos (1, 2, 4, 8, 16).
- Número de nodos/procesos (1, 2, 4 nodos virtuales).
- Variación del tamaño del problema (N pequeño, mediano y masivo).
- **Cálculo formal de métricas:**
- Ganancia de velocidad:  $S_p = \frac{T_1}{T_p}$
- Eficiencia computacional:  $E_p = \frac{S_p}{p}$
- Ajuste analítico con la **Ley de Amdahl** (Escalabilidad Fuerte) y la **Ley de Gustafson** (Escalabilidad Débil).

# 6. ESTRUCTURA DE LOS ENTREGABLES OBLIGATORIOS

## 6.1. Entregable A: Informe Técnico de Ingeniería (Formato Paper IEEE - PDF)

El documento formal no debe superar las 10 páginas y debe contener:

1. **Resumen y Abstract:** Síntesis del problema, metodología y aceleración lograda.

2. **Introducción y Contextualización Regional:** Justificación del problema en Yacuiba / Chaco.
3. **Marco Teórico y Estado del Arte:** Taxonomía de Flynn del sistema, modelo de memoria y redes de interconexión.
4. **Diseño de la Arquitectura e Infraestructura:** Diagrama del clúster/red, asignación de CPU/RAM en las PCs Dell OptiPlex y configuración de contenedores.
5. **Implementación Algorítmica Paralela:** Explicación técnica de funciones OpenMP/MPI clave, particionamiento de datos y balanceo de carga.
6. **Resultados Experimentales y Análisis de Rendimiento:** Tablas comparativas de tiempos de ejecución, gráficas de *Speedup*, Eficiencia, análisis de cuellos de botella (memoria, comunicación) y validación de las leyes de Amdahl/Gustafson.
7. **Conclusiones e Impacto Social/Productivo.**
8. **Referencias Bibliográficas.**

## 6.2. Entregable B: Repositorio de Código y Reproducibilidad (GitHub / GitLab)

El repositorio debe estar ordenado bajo la siguiente estructura de carpetas:

proyecto-paralelos-equipoXX/

```
|— README.md # Instrucciones de compilación, ejecución y topología
|— infrastructure/ # Dockerfile, docker-compose.yml o scripts de instalación
|— src/
| |— sequential/ # Código secuencial base para comparativas
| |— parallel/ # Código fuente paralelo (OpenMP / MPI / CUDA)
|— data/ # Datasets de prueba o scripts generadores de datos
|— scripts/ # Shell scripts (.sh) para automatizar el benchmarking
|— docs/ # Informe final en formato PDF
```

**Criterio estricto de reproducibilidad:** El evaluador debe poder clonar el repositorio, ejecutar un solo comando (ej. `./benchmark.sh` o `docker compose up`) y reproducir la compilación, ejecución y salida de métricas.

## 6.3. Entregable C: Defensa Técnica y Demostración en Vivo (*Live Demo*)

- Cada equipo dispondrá de **20 minutos** de defensa: 10 minutos de exposición técnica y 10 minutos de demostración en terminal e interpelación docente.
- El docente realizará preguntas individuales y solicitará modificaciones en vivo (por ejemplo, cambiar la política de schedule, modificar el número de procesos MPI o inspeccionar los hilos con herramientas del sistema como `htop` o `top`).

## 7. MATRIZ DE EVALUACIÓN ANALÍTICA (FBC - TOTAL: 100 PUNTOS)

| Dimensión de la Competencia                            | Criterio de Desempeño                                                                                                                                     | Indicadores de Evaluación                                                                                                                                                                                                                                                                                                                   | Puntaje              |
|--------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------|
| <p>SABER</p> <p>(Conceptual)</p> <p>25%</p>            | <p><b>Fundamentación Teórica y Arquitectónica:</b></p> <p>Domina los modelos de memoria, leyes de rendimiento y taxonomía del hardware utilizado.</p>     | <ul style="list-style-type: none"> <li>• Explica rigurosamente el impacto de la Ley de Amdahl y Gustafson en su proyecto.</li> <li>• Identifica con precisión los cuellos de botella de red y memoria de la arquitectura Dell OptiPlex utilizada.</li> <li>• Responde con solidez conceptual durante la defensa oral individual.</li> </ul> | <p><b>25 pts</b></p> |
| <p>SABER HACER 1</p> <p>(Procedimental)</p> <p>30%</p> | <p><b>Diseño e Implementación de Código Paralelo:</b></p> <p>Desarrolla software concurrente eficiente, libre de condiciones de carrera y optimizado.</p> | <ul style="list-style-type: none"> <li>• Paralelización efectiva y correcta mediante OpenMP, MPI o esquema híbrido.</li> <li>• Uso justificado de cláusulas de sincronización, particionamiento y reducción.</li> <li>• Verificación empírica de corrección matemática frente al modelo secuencial.</li> </ul>                              | <p><b>30 pts</b></p> |

| <b>Dimensión de la Competencia</b>                     | <b>Criterio de Desempeño</b>                                                                                                                                   | <b>Indicadores de Evaluación</b>                                                                                                                                                                                                                                                                                                      | <b>Puntaje</b>       |
|--------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------|
| <p>SABER HACER 2</p> <p>(Procedimental)</p> <p>25%</p> | <p><b>Infraestructura, Benchmarking y Reproducibilidad:</b></p> <p>Conteneriza el entorno y mide rigurosamente el rendimiento alcanzado.</p>                   | <ul style="list-style-type: none"> <li>• Despliegue funcional del clúster virtual/contenerizado (Docker/Slurm) en el laboratorio.</li> <li>• Obtención de gráficas experimentales de Speedup, Eficiencia y Escalabilidad.</li> <li>• Repositorio en GitHub 100% reproducible mediante scripts automatizados.</li> </ul>               | <p><b>25 pts</b></p> |
| <p>SABER SER</p> <p>(Actitudinal)</p> <p>20%</p>       | <p><b>Pertinencia Regional, Innovación y Trabajo en Equipo:</b></p> <p>Aplica la solución a un problema real de Yacuiba/Chaco con ética y profesionalismo.</p> | <ul style="list-style-type: none"> <li>• La temática elegida aborda una necesidad productiva, ambiental o social tangible de la región.</li> <li>• Demuestra pensamiento crítico e iniciativa para superar restricciones de hardware.</li> <li>• Trabajo colaborativo articulado y defensa técnica profesional del equipo.</li> </ul> | <p><b>20 pts</b></p> |

| Dimensión de la Competencia | Criterio de Desempeño | Indicadores de Evaluación | Puntaje |
|-----------------------------|-----------------------|---------------------------|---------|
|                             |                       |                           |         |
| TOTAL                       |                       |                           | 100 pts |

## 8. CALENDARIO DE ENTREGAS Y CONTROL DE HITOS

- **Hito 1 (Fin de Semana 2):** Presentación del Perfil Técnico (1 página: descripción del problema regional, modelo matemático y arquitectura de clúster a desplegar).
- **Hito 2 (Fin de Semana 5):** Primera revisión de avance en laboratorio: Demostración del clúster containerizado operativo en las PCs Dell y ejecución del código secuencial base.
- **Hito 3 (Fin de Semana 8):** Cierre del proyecto: Entrega del repositorio de GitHub, entrega del informe técnico en PDF y defensa oral presencial en laboratorio.