

# SISTEMAS PARALELOS

---

Clase 1 – Metodología

Prof. Enzo Rucci



FACULTAD DE INFORMATICA



UNIVERSIDAD  
NACIONAL  
DE LA PLATA

# Objetivos

- Plantear los fundamentos del procesamiento paralelo
- Caracterizar las arquitecturas de hardware
- Describir los modelos de programación
- Estudiar métricas de rendimiento
- Analizar y trabajar casos concretos de procesamiento paralelo, resolubles sobre distintas arquitecturas multiprocesador.
- Analizar tendencias y avances futuros en lo referido al procesamiento paralelo

# Objetivos y resultados esperados

- Durante la cursada, aprenderán sobre arquitecturas, modelos y librerías de programación específicas, entre otros.
- Sin embargo, se apunta a que aprendan una manera de pensar orientada a hacer un mejor aprovechamiento del uso de recursos y a obtener eficiencia en el procesamiento.
- Se espera que puedan encontrar utilidad a esta manera de pensar para que puedan aplicar en cualquier problema a resolver.

# Equipo docente

- Prof. Dr Enzo Rucci
- JTP Dr Adrián Pousa
- Auxiliares:
  - Ing. Leandro Libutti
  - Ing. Lautaro Gesuelli

# Metodología

- Clases de teoría
  - Viernes a las 8:30hs. Sala de PC de Postgrado (2do piso).
  - Se pondrá a disposición de los alumnos material audiovisual, además de las diapositivas habituales.
- Explicaciones y consulta de práctica
  - Viernes de 10 a 12hs. Sala de PC - Postgrado. Inicio: 17/03.
  - Martes de 17 a 19hs (Webex). Inicio 14/03.
    - [URL:https://unlpeducar.webex.com/unlpeducar-sp/j.php?MTID=m58e6717c4dc79282174ad4d654371117](https://unlpeducar.webex.com/unlpeducar-sp/j.php?MTID=m58e6717c4dc79282174ad4d654371117)
    - También se podrán hacer consultas por medio de la plataforma de Educación a Distancia Ideas en el resto de la semana. En los casos que el docente considere necesario, se responderá la consulta en el encuentro sincrónico o presencial.

# Metodología

- Aprobación de la cursada:
  - A partir de la entrega y aprobación de 3 trabajos prácticos (TPs) y sus respectivos coloquios (obligatorios). Fecha recuperatoria.
  - Los TPs deben desarrollarse en grupos de 2 personas, aunque los coloquios serán individuales
    - → Ver Cronograma
    - → Enviar conformación de grupos por mensajería de IDEAS.
  - El trabajo experimental debe realizarse en equipo remoto de la cátedra (salvo que se indique lo contrario).
  - Se recomienda “desarrollar en casa” y usar el tiempo de consulta para despejar dudas.

# Metodología

- Requisito de asistencia
  - Ni las clases teóricas ni las clases prácticas son de asistencia obligatoria. De todas formas, la cátedra recomienda ampliamente asistir para un mejor aprendizaje de los contenidos considerando la fuerte vinculación entre las clases teóricas y las prácticas.
  - Los coloquios de las entregas sí son obligatorios, ya que forman parte del mecanismo de evaluación para la aprobación de la cursada. Aquellos alumnos que requieran un certificado de examen, pueden solicitarlo al JTP.

# Metodología

- Aprobación del final:
  - Aprobar un parcial teórico al final de la cursada → en caso de baja nota, recuperación a partir de desarrollo de un trabajo monográfico dado por la cátedra y su posterior coloquio en mesa de final (hasta mesa de Marzo de 2024)].
  - Rendir examen final escrito en alguna mesa de examen.
  - Desarrollo y prueba de soluciones paralelas para un problema concreto + elaboración de informe de análisis de rendimiento + coloquio de defensa



# Metodología

- Material y comunicación: Plataforma Ideas ([ideas.info.unlp.edu.ar](http://ideas.info.unlp.edu.ar))
  - Buscar curso «Sistemas Paralelos 2023» y solicitar inscripción
- Correo de la cátedra: [sparalelos@lidi.info.unlp.edu.ar](mailto:sparalelos@lidi.info.unlp.edu.ar)

# Bibliografía básica

- “Introduction to Parallel Computing”. Grama, Gupta, Karypis, Kumar. Addison Wesley (2003).
- “Introduction to High Performance Computing for Scientists and Engineers”. Georg Hager, Gerard Wellein. CRC Press (2011).
- “Parallel Programming for Multicore and Cluster Systems”. Thomas Rauber, Gudulla Runger. Springer (2010).
- “An introduction to parallel programming”. Peter Pacheco. Elsevier (2011).
- “Parallel Programming”. Wilkinson, Allen. Prentice Hall 2005.