

# Programación para Física y Astronomía

Departamento de Física.

---

Corodinadora: C Loyola

Profesores C Femenías / F Bugini / D Basantes

Primer Semestre 2025

Universidad Andrés Bello

Departamento de Física y Astronomía



Introducción y Contexto

Retroalimentación Individual

Proyectos Finales

Refuerzo de Temas Pendientes

Conclusiones y Próximos Pasos

# Introducción y Contexto

---

# Introducción y Contexto ∈ Contexto Tras la Solemne II (Continuación)

- **Semana 11, Sesión 1 (Sesión 21):**
  - Retroalimentación general de la Solemne II.
  - Profundización en **POO avanzada**: polimorfismo, composición.
  - Propuesta de ejemplos físicos (Body, Star, SolarSystem).
- **Semana 11, Sesión 2 (Sesión 22):**
  - Podríamos comenzar a plantear **proyectos colaborativos** o la fase final del curso, dependiendo del Syllabus.
  - Seguir revisando retroalimentación **individual** y plan de recuperación si alguien necesita.
- **Objetivo de hoy:** Organizar ideas para trabajos/proyectos finales, o ejercicios de refuerzo según la ruta del Syllabus.

- **Compartir** retroalimentación más detallada de la Solemne II a nivel individual (cuando corresponda).
- **Proponer** líneas de trabajo para un **proyecto integrador** (o mini-proyecto), combinando POO y análisis de datos.
- **Fomentar** la discusión y asignación de **equipos** o **temáticas**, si es un proyecto grupal.
- **Refrescar** cualquier duda sobre polimorfismo, composición u otros temas recientes.

# Retroalimentación Individual

---

- **Notas y comentarios** ya publicadas en CANVAS.
- Revisen su **archivo corregido** (o PDF de correcciones) para ver comentarios específicos.
- **Pregunten** si hay dudas sobre la calificación o la forma de mejorar.

- Medias y desv. estándar de notas, si corresponde.
- Observaciones sobre **tiempo** y **estrategia** de resolución.
- Tipos de errores frecuentes:
  - Falta de **super()** en herencia.
  - Errores de **reshape** en NumPy o confusión en subplots.
  - Falta de validación en **pandas** (CSV mal formateados).



# Proyectos Finales

---

## Idea General

- **Objetivo:** desarrollar un mini-proyecto donde se combinen:
  - **POO** para modelar entidades (partículas, cuerpos astronómicos, datos experimentales).
  - **NumPy** para cálculos numéricos (simulaciones, estadísticas).
  - **Matplotlib** para visualización 2D/3D.
  - (Opcional) **pandas** para manejo de datos más robusto (CSV, merges, etc.).
- **Equipo:** 2-3 personas, entregando un **repositorio** o un **notebook** en Colab con explicación.
- **Extensión:** 1-2 semanas de desarrollo, con check-ins intermedios.

- **Simulación de órbitas** en un sistema planetario simple (**Body**, **SolarSystem**).
- **Modelo de cargas** (**Particle**, **ChargedParticle**) e interacción electrostática.
- **Análisis de datos experimentales** con una clase **Measurement** y comparaciones estadísticas en **pandas**.
- **Animaciones** con Matplotlib (**FuncAnimation**) de un movimiento 2D/3D.
- Cualquier otra idea relevante al campo de Física/Astronomía/Ingeniería.

- **Formación de grupos:** 2-3 integrantes.
- **Plan de trabajo:**
  - Propuesta inicial (alcances, clases a definir, datos a usar).
  - Avance parcial (estructuras POO, prototipos de gráficas).
  - Entrega final (documentación en un **README** o en celdas de **Markdown**).
- **Evaluación:**
  - Organización y claridad del código (POO, funciones auxiliares).
  - Uso correcto de NumPy/Matplotlib/pandas según corresponda.
  - Resultados coherentes, gráficas interpretables.
  - Documentación y conclusiones (Markdown o PDF anexo).

## Refuerzo de Temas Pendientes

---

## Refuerzo de Temas Pendientes ∈ Recopilación de Dudas Pendientes

- **POO avanzada:** polimorfismo, composición, encapsulación relativa en Python.
- **NumPy:** broadcast, random, performance tips.
- **Matplotlib:** subplots 3D, animación (FuncAnimation), personalización avanzada (fonts, styles).
- **pandas:** merges, groupby, time-series (si aplica).

Comparte tus inquietudes en voz alta o en foros.

## Refuerzo de Temas Pendientes $\in$ Minicharla: Animaciones (Opcional)

- Usar `matplotlib.animation.FuncAnimation`.
- Estructura típica:

```
1  import matplotlib.animation as animation
2
3  fig, ax = plt.subplots()
4  line, = ax.plot([], [], 'o-')
5
6  def init():
7      ax.set_xlim(0, 10)
8      ax.set_ylim(-1, 1)
9      return line,
10
11  def update(frame):
12      xdata.append(frame)
13      ydata.append(np.sin(frame))
14      line.set_data(xdata, ydata)
15      return line,
16
```

## Conclusiones y Próximos Pasos

---



- ¿Qué proyecto integrador les interesa más?
- ¿Necesitan datos reales (CSV) o preferirían algo 100% simulado (NumPy random)?
- ¿Prefieren foco en **POO** o en **análisis/visualización**?
- **Objetivo** de la discusión: definir **lineamientos** finales del proyecto (o mini-tarea).

- Continuamos con la retroalimentación de la Solemne II, ahora a nivel **individual** y con consejos puntuales.
- Iniciamos la **planificación** de proyectos finales o colaborativos, unificando POO y análisis de datos.
- Reforzamos la disponibilidad de **espacios de dudas** para temas pendientes (polimorfismo, animaciones, etc.).

- **Semana 12:** Puesta en marcha de proyectos (si corresponde) o siguientes unidades del Syllabus.
- Revisar la **fechas** de entrega intermedia (si se establece).
- Seguir explorando **animaciones, estadística avanzada, o paralelización** (según plan).

- **Real Python** - Sección de proyectos y guías prácticas.
- **Matplotlib Animation API** - documentación oficial.
- **NumPy random** - generadores pseudoaleatorios avanzados.
- **Canvas/Foros**: para la comunicación de consultas y feedback continuo.

# ente trabajo y hasta la próxima s

- Consulten en foros o mail si tienen dudas detalladas del proyecto/propuesta.
- ¡Nos vemos en Semana 12 con más desarrollo de ideas!