



Profesor:

Michael Heredia Pérez
mherediap@unal.edu.co

Taller de repaso de prerequisitos: cálculos, álgebra lineal y programación.

Información importante

- Este taller es de **carácter autónomo y no es calificable**. Su objetivo es apoyar el estudio personal de los prerequisitos necesarios para cursar Mecánica de Sólidos y Mecánica Tensorial.
- No es obligatorio resolver completamente este taller. Se recomienda utilizarlo como una **herramienta de autoevaluación**, verificando si comprende cada pregunta, si reconoce los conceptos involucrados y si sería capaz de desarrollar una respuesta razonada.
- En caso de decidir resolverlo, se recomienda hacerlo **a mano**, priorizando esquemas, ecuaciones clave y razonamientos, más que respuestas extensas. Alternativamente, puede utilizar herramientas digitales como *Obsidian*^a, *Notion*^b u otras plataformas que soporten **Markdown** y escritura de fórmulas matemáticas.
- El estudiante debe tener en cuenta el material complementario disponible en el **repositorio del curso en GitHub**^c, donde se encuentran enlaces, guías, videos y recursos adicionales para profundizar en los temas tratados.
- Se invita explícitamente a utilizar herramientas de **inteligencia artificial** como apoyo al estudio, siempre con un enfoque formativo: para aclarar conceptos, obtener explicaciones alternativas o entender procedimientos matemáticos y computacionales, y no como un sustituto del razonamiento propio.
- Al responder las preguntas, procure siempre preguntarse si entiende el concepto, el procedimiento matemático, la interpretación física y, cuando aplique, la lógica del código. El objetivo es **comprender**, no solo obtener un resultado.
- Aunque este taller no es evaluable, **todos los temas aquí tratados hacen parte de los prerequisitos del curso**. Si un tema no aparece explícitamente en este taller pero fue visto durante el estudio autónomo o en cursos previos, sigue siendo susceptible de ser evaluado.

^a<https://obsidian.md/>

^b<https://www.notion.com/>

^c<https://github.com/michaelherediaperez/medio-continuo>

Índice

1. Programación de computadores en Python	3
1.1. Tipos de datos	3
1.2. Estructuras de control	3
1.3. Uso de librerías en Python	3
1.4. Librería <code>numpy</code> (cálculo algebraico)	4
1.5. Librería <code>sympy</code> (cálculo simbólico)	4
1.6. Librería <code>matplotlib</code> (graficación)	4
1.7. Jupyter Notebook	5
2. Álgebra lineal	5
2.1. Norma de un vector	5
2.2. Producto punto	5
2.3. Producto cruz	6
2.4. Proyección de vectores	6
2.5. Cosenos directores	6
2.6. Matrices	7
2.7. Determinantes	7
2.8. Espacios vectoriales	7
2.9. Dependencia e independencia lineal	7
2.10. Bases	8
2.11. Planos y líneas rectas	8
3. Cálculo vectorial	8
3.1. Campo vectorial	8
3.2. Gradiente	9
3.3. Matriz jacobiana y jacobiano	9
3.4. Divergencia	9
3.5. Rotacional	9
3.6. Diferenciales	10
3.7. Expansión en series de Taylor	10
3.8. Regla de la cadena	10
4. Estática	11
4.1. Equilibrio de cuerpos bidimensionales	11
4.2. Equilibrio de cuerpos tridimensionales	11
4.3. Momento de inercia	11
4.4. Momento flector	12
4.5. Fuerza cortante	12
4.6. Conceptos aplicados a vigas	12
5. Ecuaciones diferenciales	13
5.1. Concepto de ecuación diferencial	13
5.2. Ecuaciones diferenciales ordinarias	13
5.3. Problema homogéneo y problema completo	13
5.4. Sistemas dinámicos	14
5.5. Ecuaciones diferenciales parciales	14

1. Programación de computadores en Python

La programación en Python se utiliza en este curso como una herramienta de apoyo para explorar conceptos matemáticos y físicos, realizar cálculos simbólicos y numéricos, y visualizar resultados. No se espera que el estudiante sea programador, pero sí que entienda la lógica básica del código y su relación con los procedimientos matemáticos. Un manejo mínimo de Python facilita el análisis de campos, operadores diferenciales y sistemas de ecuaciones que aparecen de forma natural en la mecánica del medio continuo.

1.1. Tipos de datos

1. **Pregunta conceptual.** ¿Qué es un tipo de dato en Python? ¿Entiende la diferencia entre un número entero, un número flotante y un número complejo? ¿Puede explicar qué es una cadena de texto y para qué se usa? ¿Entiende qué es una lista y en qué se diferencia de una tupla?
2. **Ejercicio práctico.** Defina en Python variables de los siguientes tipos:

- `int, float, complex`
- `str`
- `list y tuple`

Imprima su valor y su tipo usando `type()`. ¿Entiende por qué Python asigna cada tipo?

3. **Ejercicio reflexivo.** Modifique una lista y luego intente modificar una tupla. ¿Qué ocurre? ¿Entiende el concepto de mutabilidad?

1.2. Estructuras de control

1. **Pregunta conceptual.** ¿Qué es una estructura de control? ¿Entiende para qué sirven las sentencias `if`, `for` y `while`? ¿Puede explicar la diferencia entre un bucle `for` y uno `while`?
2. **Ejercicio práctico.** Escriba un programa que:

- Recorra una lista de números.
- Determine cuáles son positivos, negativos o cero.
- Imprima un mensaje para cada caso.

¿Entiende la lógica condicional implementada?

3. **Ejercicio computacional.** Implemente un bucle que calcule la suma de los primeros n números naturales. ¿Entiende cómo se controla el flujo del programa?

1.3. Uso de librerías en Python

1. **Pregunta conceptual.** ¿Qué es una librería en Python? ¿Entiende la diferencia entre `import modulo` y `from modulo import objeto`? ¿Puede explicar por qué es buena práctica usar librerías externas?
2. **Ejercicio práctico.** Importe las siguientes librerías:

- `math`
- `numpy`
- `sympy`

Use al menos una función de cada una. ¿Entiende qué problema resuelve cada librería?

3. **Ejercicio reflexivo.** Compare calcular una raíz cuadrada con y sin librerías. ¿Entiende las ventajas del uso de librerías?

1.4. Librería `numpy` (cálculo algebraico)

1. **Pregunta conceptual.** ¿Qué es `numpy`? ¿Entiende la diferencia entre una lista de Python y un arreglo de `numpy`? ¿Puede explicar por qué `numpy` es fundamental para cálculo numérico?
2. **Ejercicio práctico.** Defina:
 - Un vector usando `numpy`.
 - Una matriz usando `numpy`.

Realice operaciones básicas (suma, producto, norma). ¿Entiende cómo estas operaciones representan conceptos de álgebra lineal?

3. **Ejercicio computacional.** Compare el tiempo de ejecución entre listas y arreglos `numpy`. ¿Entiende por qué `numpy` es más eficiente?

1.5. Librería `sympy` (cálculo simbólico)

1. **Pregunta conceptual.** ¿Qué es el cálculo simbólico? ¿Entiende la diferencia entre cálculo simbólico y numérico? ¿Puede explicar cuándo es conveniente usar `sympy`?
2. **Ejercicio práctico.** Defina una variable simbólica y una función:
 - Calcule derivadas.
 - Calcule integrales.
 - Simplifique expresiones.

¿Entiende el significado matemático de cada operación?

3. **Ejercicio computacional.** Use `sympy` para calcular un gradiente o una matriz jacobiana sencilla. ¿Entiende cómo el código refleja la derivación analítica?

1.6. Librería `matplotlib` (graficación)

1. **Pregunta conceptual.** ¿Qué es graficar un conjunto de datos? ¿Entiende la diferencia entre una gráfica y una visualización de información? ¿Puede explicar por qué las gráficas son fundamentales en ingeniería?
2. **Ejercicio práctico.** Grafique:
 - Una función $f(x) = x^2$.
 - Un campo vectorial simple.

¿Entiende qué representa cada eje y cada elemento gráfico?

3. **Ejercicio computacional.** Modifique títulos, etiquetas y estilos de la gráfica. ¿Entiende cómo mejorar la claridad de una visualización?

1.7. Jupyter Notebook

1. **Pregunta conceptual.** ¿Qué es un Jupyter Notebook? ¿Entiende la diferencia entre un archivo .py y un notebook? ¿Puede explicar por qué Jupyter es útil para aprendizaje y análisis?
2. **Ejercicio práctico.** Cree un notebook que:
 - Incluya texto en Markdown.
 - Contenga celdas de código en Python.
 - Ejecute cálculos y gráficas.

¿Entiende la diferencia entre celdas de código y celdas de texto?
3. **Ejercicio reflexivo.** Reordene las celdas y vuelva a ejecutar el notebook. ¿Qué ocurre? ¿Entiende el estado de ejecución de un notebook?

2. Álgebra lineal

El álgebra lineal es una base fundamental de la mecánica del medio continuo, ya que gran parte de sus variables y ecuaciones se expresan en términos de vectores, matrices y transformaciones lineales. Conceptos como producto punto, producto cruz, bases y valores propios aparecen de manera directa en la descripción de tensores, cambios de sistema de referencia y análisis de direcciones principales. Comprender estos elementos permite interpretar correctamente las formulaciones tensoriales que se desarrollan más adelante en el curso.

2.1. Norma de un vector

1. **Pregunta conceptual.** ¿Qué es la norma de un vector \mathbf{v} ? ¿Entiende la norma como una longitud geométrica o solo como una fórmula algebraica? ¿Puede explicar por qué la norma siempre es un número no negativo? ¿Entiende qué significa que $\|\mathbf{v}\| = 0$?
2. **Ejercicio práctico.** Dados los vectores $\mathbf{v}_1 = (3, 4)$ y $\mathbf{v}_2 = (1, -2, 2)$:
 - Calcule sus normas.
 - Verifique si alguno es unitario.
 - Interprete el resultado geométricamente.

¿Entiende cada paso del procedimiento?
3. **Ejercicio computacional.** Calcule las normas anteriores usando Python. ¿Entiende cómo el código implementa la definición matemática de norma?

2.2. Producto punto

1. **Pregunta conceptual.** ¿Qué es el producto punto entre dos vectores $\mathbf{u} \cdot \mathbf{v}$? ¿Entiende su definición algebraica y su interpretación geométrica? ¿Puede explicar qué ocurre cuando el producto punto es cero? ¿Reconoce todas sus propiedades (comutatividad, bilinealidad, relación con el ángulo)?
2. **Ejercicio práctico.** Dados $\mathbf{u} = (1, 2, 3)$ y $\mathbf{v} = (2, -1, 1)$:
 - Calcule $\mathbf{u} \cdot \mathbf{v}$.
 - Determine el ángulo entre los vectores.

- Analice si los vectores son ortogonales.

¿Entiende qué información física entrega el ángulo obtenido?

3. **Ejercicio computacional.** Implemente el cálculo del producto punto y del ángulo en Python. ¿Entiende cómo se relacionan las operaciones del código con las expresiones matemáticas?

2.3. Producto cruz

1. **Pregunta conceptual.** ¿Qué es el producto cruz $\mathbf{u} \times \mathbf{v}$? ¿Entiende por qué solo está definido en \mathbb{R}^3 ? ¿Puede explicar su interpretación geométrica en términos de área y dirección? ¿Reconoce sus propiedades algebraicas y su carácter no conmutativo?

2. **Ejercicio práctico.** Dados $\mathbf{u} = (1, 0, 1)$ y $\mathbf{v} = (0, 1, 1)$:

- Calcule $\mathbf{u} \times \mathbf{v}$.
- Determine el área del paralelogramo generado.
- Verifique la ortogonalidad del resultado.

¿Entiende por qué el vector resultante es perpendicular al plano generado?

3. **Ejercicio computacional.** Grafique en Python los vectores y su producto cruz. ¿Entiende la relación entre el resultado numérico y la visualización?

4. **Pregunta bonus.** ¿Puede nombrar las más de 10 propiedades que tiene esta operación?

2.4. Proyección de vectores

1. **Pregunta conceptual.** ¿Qué significa proyectar un vector sobre otro? ¿Entiende la proyección como una descomposición geométrica? ¿Puede explicar la relación entre proyección y producto punto?

2. **Ejercicio práctico.** Proyecte $\mathbf{u} = (2, 3)$ sobre $\mathbf{v} = (1, 1)$:

- Calcule la proyección vectorial.
- Calcule la proyección escalar.
- Interprete los resultados.

¿Entiende la diferencia entre ambas proyecciones?

3. **Ejercicio computacional.** Implemente el cálculo de la proyección en Python. ¿Entiende por qué es necesario normalizar el vector?

2.5. Cosenos directores

1. **Pregunta conceptual.** ¿Qué son los cosenos directores de un vector? ¿Entiende su relación con la orientación del vector en el espacio? ¿Puede explicar por qué la suma de los cuadrados de los cosenos directores es igual a 1?

2. **Ejercicio práctico.** Calcule los cosenos directores del vector $\mathbf{v} = (2, -1, 2)$. ¿Puede interpretar el signo y magnitud de cada coseno?

3. **Ejercicio computacional.** Calcule los cosenos directores usando Python. ¿Entiende cómo se relacionan con la normalización del vector?

2.6. Matrices

1. **Pregunta conceptual.** ¿Qué es una matriz desde el punto de vista algebraico y geométrico? ¿Entiende una matriz como un operador lineal? ¿Puede explicar la diferencia entre una matriz y un sistema de ecuaciones?

2. **Ejercicio práctico.** Dada la matriz

$$\mathbf{A} = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix},$$

- Aplique \mathbf{A} a un vector genérico.
- Interprete el resultado.

¿Entiende cómo la matriz transforma el espacio?

3. **Ejercicio computacional.** Visualice la acción de \mathbf{A} sobre varios vectores en Python. ¿Entiende el efecto global de la transformación?

2.7. Determinantes

1. **Pregunta conceptual.** ¿Qué es el determinante de una matriz? ¿Entiende su significado geométrico? ¿Puede explicar qué implica que el determinante sea cero?

2. **Ejercicio práctico.** Calcule el determinante de

$$\mathbf{A} = \begin{pmatrix} 2 & 1 \\ 4 & 2 \end{pmatrix}.$$

¿Es invertible la matriz? ¿Por qué?

3. **Ejercicio computacional.** Calcule el determinante en Python y compare con el resultado manual. ¿Entiende la coherencia entre ambos resultados?

2.8. Espacios vectoriales

1. **Pregunta conceptual.** ¿Qué es un espacio vectorial? ¿Entiende la importancia de los axiomas? ¿Puede dar ejemplos y contraejemplos?

2. **Ejercicio práctico.** Determine si los siguientes conjuntos son espacios vectoriales:

- \mathbb{R}^2
- El conjunto de polinomios de grado menor o igual a 2
- El conjunto de vectores con norma 1

¿Entiende qué axioma falla cuando no lo son?

2.9. Dependencia e independencia lineal

1. **Pregunta conceptual.** ¿Qué significa que un conjunto de vectores sea linealmente independiente? ¿Entiende la relación con redundancia de información?

2. **Ejercicio práctico.** Determine si los vectores $\mathbf{v}_1 = (1, 2)$, $\mathbf{v}_2 = (2, 4)$ y $\mathbf{v}_3 = (1, 0)$ son linealmente independientes. ¿Puede justificar su respuesta algebraica y geométricamente?

3. **Ejercicio computacional.** Verifique la independencia usando Python. ¿Entiende el criterio usado por el algoritmo?

2.10. Bases

1. **Pregunta conceptual.** ¿Qué es una base de un espacio vectorial? ¿Entiende por qué una base genera todo el espacio? ¿Puede explicar la relación entre base y coordenadas?
2. **Ejercicio práctico.** Determine si los vectores $\{(1, 0), (1, 1)\}$ forman una base de \mathbb{R}^2 . Exprese un vector cualquiera en esa base.
3. **Ejercicio computacional.** Realice el cambio de base en Python. ¿Entiende el procedimiento matemático detrás del código?

2.11. Planos y líneas rectas

1. **Pregunta conceptual.** ¿Qué es una línea recta y qué es un plano en \mathbb{R}^3 ? ¿Entiende las distintas formas de representación (paramétrica, cartesiana, vectorial)?
2. **Ejercicio práctico.** Encuentre la ecuación del plano que pasa por tres puntos dados. Determine si un punto pertenece al plano.
3. **Ejercicio computacional.** Grafique el plano y la línea en Python. ¿Entiende la relación entre las ecuaciones y la visualización?

3. Cálculo vectorial

El cálculo vectorial proporciona el lenguaje matemático necesario para describir campos físicos que varían en el espacio, como los campos de desplazamiento, deformación y esfuerzo. Operadores como el gradiente, la divergencia y el rotacional aparecen de forma natural en las ecuaciones de equilibrio y compatibilidad de la mecánica del medio continuo. Un manejo claro de estos conceptos es indispensable para comprender la formulación diferencial de los problemas en mecánica de sólidos y mecánica de fluidos.

3.1. Campo vectorial

1. **Pregunta conceptual.** ¿Qué es un campo vectorial? ¿Entiende la diferencia entre un campo escalar y un campo vectorial? ¿Puede explicar qué representa físicamente un campo vectorial en ingeniería (por ejemplo, velocidad, fuerza o flujo)? ¿Entiende el dominio y la imagen de un campo vectorial?
2. **Ejercicio práctico.** Considere el campo vectorial $\mathbf{F}(x, y) = (x, y)$:
 - Evalúe el campo en distintos puntos.
 - Describa cualitativamente su comportamiento.
 - Interprete físicamente el campo.¿Entiende cómo cambia el vector según la posición?
3. **Ejercicio computacional.** Grafique el campo vectorial en Python. ¿Entiende cómo el gráfico representa la definición matemática?

3.2. Gradiente

1. **Pregunta conceptual.** ¿Qué es el gradiente de un campo escalar? ¿Entiende al gradiente como un vector o como un operador? ¿Puede explicar su significado geométrico y físico? ¿Entiende por qué el gradiente apunta en la dirección de máximo crecimiento?

2. **Ejercicio práctico.** Dado el campo escalar $f(x, y) = x^2 + 2y^2$:

- Calcule ∇f .
- Evalúe el gradiente en $(1, 1)$.
- Interprete el resultado.

¿Entiende qué información local entrega el gradiente?

3. **Ejercicio computacional.** Calcule el gradiente usando Python. ¿Entiende la diferencia entre evaluar simbólicamente y numéricamente?

3.3. Matriz jacobiana y jacobiano

1. **Pregunta conceptual.** ¿Qué es la matriz jacobiana de una función vectorial? ¿Entiende la diferencia entre la matriz jacobiana y el jacobiano (determinante)? ¿Puede explicar su significado como aproximación lineal local? ¿Entiende su interpretación geométrica como cambio de escala y deformación?

2. **Ejercicio práctico.** Dada la función vectorial $\mathbf{F}(x, y) = (x^2y, y^2)$:

- Calcule la matriz jacobiana.
- Calcule el jacobiano.
- Interprete ambos resultados.

¿Entiende qué información aporta cada uno?

3. **Ejercicio computacional.** Calcule la matriz jacobiana y su determinante en Python. ¿Entiende cómo el código refleja la derivación matemática?

3.4. Divergencia

1. **Pregunta conceptual.** ¿Qué mide la divergencia de un campo vectorial? ¿Entiende la divergencia como una medida local de fuente o sumidero? ¿Puede explicar su significado físico en términos de flujo?

2. **Ejercicio práctico.** Dado el campo $\mathbf{F}(x, y) = (x, -y)$:

- Calcule $\nabla \cdot \mathbf{F}$.
- Interprete el resultado en distintos puntos.

¿Entiende por qué la divergencia puede ser positiva, negativa o nula?

3. **Ejercicio computacional.** Visualice el campo y analice su divergencia en Python. ¿Entiende la coherencia entre el resultado analítico y el comportamiento gráfico?

3.5. Rotacional

1. **Pregunta conceptual.** ¿Qué es el rotacional de un campo vectorial? ¿Entiende su relación con el concepto de giro local? ¿Puede explicar por qué el rotacional es un vector? ¿Entiende su interpretación

física en problemas de mecánica de fluidos o sólidos?

2. **Ejercicio práctico.** Dado el campo $\mathbf{F}(x, y) = (-y, x)$:

- Calcule $\nabla \times \mathbf{F}$.
- Interprete el resultado.

¿Entiende por qué el campo presenta rotación?

3. **Ejercicio computacional.** Grafique el campo vectorial y analice su rotacional en Python. ¿Entiende la relación entre la visualización y el concepto matemático?

3.6. Diferenciales

1. **Pregunta conceptual.** ¿Qué es un diferencial? ¿Entiende el diferencial como una aproximación lineal?

¿Puede explicar la diferencia entre diferencial y derivada?

2. **Ejercicio práctico.** Para la función $f(x, y) = x^2 + y^2$:

- Calcule el diferencial total.
- Aproxime el cambio en f cerca del punto $(1, 1)$.

¿Entiende cuándo esta aproximación es válida?

3. **Ejercicio computacional.** Implemente el cálculo del diferencial en Python. ¿Entiende la relación con la expansión de Taylor?

3.7. Expansión en series de Taylor

1. **Pregunta conceptual.** ¿Qué es una serie de Taylor? ¿Entiende la expansión como una aproximación local? ¿Puede explicar la diferencia entre el caso univariado y multivariado?

2. **Ejercicio práctico.**

- Obtenga la expansión de Taylor de $f(x) = \sin x$ alrededor de $x = 0$.
- Obtenga la expansión de $f(x, y) = x^2y$ alrededor de $(0, 0)$.

¿Entiende el significado de cada término de la expansión?

3. **Ejercicio computacional.** Calcule las expansiones en Python. ¿Entiende cómo se construye la serie término a término?

3.8. Regla de la cadena

1. **Pregunta conceptual.** ¿Qué expresa la regla de la cadena? ¿Entiende su importancia en funciones compuestas? ¿Puede explicar la regla de la cadena usando matrices jacobianas?

2. **Ejercicio práctico.** Sea $z = f(x, y)$ con $x = g(t)$ y $y = h(t)$:

- Aplique la regla de la cadena para $\frac{dz}{dt}$.
- Interprete el resultado.

¿Entiende el flujo de dependencias entre variables?

3. **Ejercicio computacional.** Implemente el cálculo usando Python. ¿Entiende cómo la estructura del código refleja la composición de funciones?

4. Estática

La estática introduce los principios de equilibrio que constituyen el punto de partida de la mecánica del medio continuo. Conceptos como fuerzas internas, momentos, cortantes y momentos flectores permiten comprender cómo se transmiten las cargas dentro de un cuerpo. Estos principios se generalizan posteriormente a nivel diferencial, dando lugar a las ecuaciones de equilibrio que gobiernan el comportamiento de medios continuos.

4.1. Equilibrio de cuerpos bidimensionales

1. **Pregunta conceptual.** ¿Qué significa que un cuerpo esté en equilibrio estático en dos dimensiones? ¿Cuáles son las ecuaciones de equilibrio y qué representa físicamente cada una? ¿Entiende por qué no basta con que la suma de fuerzas sea cero? ¿Puede explicar el rol del momento en el equilibrio?
2. **Ejercicio práctico.** Considere un cuerpo rígido en el plano sometido a varias fuerzas:
 - Dibuje el diagrama de cuerpo libre.
 - Plantee las ecuaciones $\sum F_x = 0$, $\sum F_y = 0$ y $\sum M = 0$.
 - Determine las reacciones.¿Entiende el significado físico de cada ecuación?
3. **Ejercicio computacional.** Resuelva el sistema de ecuaciones en Python. ¿Entiende cómo las ecuaciones de equilibrio se traducen en un sistema algebraico?

4.2. Equilibrio de cuerpos tridimensionales

1. **Pregunta conceptual.** ¿Qué diferencias existen entre el equilibrio en 2D y en 3D? ¿Cuántas ecuaciones de equilibrio se requieren en tres dimensiones y por qué? ¿Entiende el significado físico de momentos respecto a ejes?
2. **Ejercicio práctico.** Para un cuerpo rígido en el espacio:
 - Dibuje el diagrama de cuerpo libre.
 - Plantee las seis ecuaciones de equilibrio.
 - Identifique posibles redundancias o indeterminaciones.¿Entiende cuándo un sistema es estáticamente determinado?
3. **Ejercicio computacional.** Implemente las ecuaciones de equilibrio en Python. ¿Entiende la relación entre estática tridimensional y álgebra lineal?

4.3. Momento de inercia

1. **Pregunta conceptual.** ¿Qué es el momento de inercia geométrico? ¿Entiende su diferencia con el momento de inercia de masa? ¿Puede explicar su significado físico en flexión de vigas?
2. **Ejercicio práctico.** Calcule el momento de inercia respecto a un eje dado para:
 - Un rectángulo.

- Un círculo.

Use tanto la definición integral como fórmulas conocidas. ¿Entiende por qué el eje de referencia es importante?

3. **Ejercicio computacional.** Implemente el cálculo del momento de inercia en Python. ¿Entiende cómo la integral se aproxima numéricamente?

4.4. Momento flector

1. **Pregunta conceptual.** ¿Qué es el momento flector en una viga? ¿Entiende su relación con la curvatura de la viga? ¿Puede explicar el signo del momento flector y su convención?

2. **Ejercicio práctico.** Para una viga simplemente apoyada con carga distribuida:

- Calcule las reacciones.
- Determine la expresión del momento flector.
- Identifique el momento máximo.

¿Entiende cómo el momento varía a lo largo de la viga?

3. **Ejercicio computacional.** Grafique el diagrama de momento flector en Python. ¿Entiende la relación entre la expresión analítica y el diagrama?

4.5. Fuerza cortante

1. **Pregunta conceptual.** ¿Qué es la fuerza cortante en una viga? ¿Entiende su relación con el equilibrio diferencial? ¿Puede explicar cómo se relaciona con el momento flector?

2. **Ejercicio práctico.** Para la misma viga del ejercicio anterior:

- Determine la fuerza cortante a lo largo de la viga.
- Identifique los puntos donde cambia su valor.

¿Entiende la relación entre cargas aplicadas y discontinuidades en el diagrama?

3. **Ejercicio computacional.** Grafique el diagrama de fuerza cortante en Python. ¿Entiende la coherencia entre el diagrama de cortante y el de momento?

4.6. Conceptos aplicados a vigas

1. **Pregunta conceptual.** ¿Qué hipótesis se asumen al modelar una viga en estática? ¿Entiende las diferencias entre viga simplemente apoyada, empotrada y en voladizo? ¿Puede explicar cuándo estas idealizaciones dejan de ser válidas?

2. **Ejercicio práctico.** Compare el comportamiento de una misma carga aplicada sobre:

- Una viga simplemente apoyada.
- Una viga empotrada.

Analice diferencias en reacciones, cortantes y momentos. ¿Entiende por qué cambian los resultados?

3. **Ejercicio computacional.** Implemente ambos casos en Python y compare resultados. ¿Entiende cómo las condiciones de apoyo modifican el modelo matemático?

5. Ecuaciones diferenciales

Las ecuaciones diferenciales constituyen el núcleo matemático de la mecánica del medio continuo, ya que las leyes físicas se expresan usualmente como relaciones diferenciales entre campos. Aunque en este curso se hace un uso limitado de técnicas de solución, es fundamental que el estudiante entienda la estructura de las ecuaciones diferenciales ordinarias y parciales, el significado de sus soluciones y el papel de las condiciones iniciales y de frontera en problemas de elasticidad.

5.1. Concepto de ecuación diferencial

1. **Pregunta conceptual.** ¿Qué es una ecuación diferencial? ¿Entiende la diferencia entre una ecuación algebraica y una ecuación diferencial? ¿Puede explicar qué representa una ecuación diferencial desde el punto de vista físico? ¿Entiende qué significa el orden de una ecuación diferencial?

2. **Ejercicio práctico.** Clasifique las siguientes ecuaciones:

- $\frac{dx}{dt} + kx = 0$
- $\frac{d^2x}{dt^2} + \omega^2x = 0$
- $\frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} = 0$

¿Son ordinarias o parciales? ¿De qué orden son?

3. **Ejercicio reflexivo.** Identifique variables independientes y dependientes en cada caso. ¿Entiende por qué esta distinción es fundamental?

5.2. Ecuaciones diferenciales ordinarias

1. **Pregunta conceptual.** ¿Qué caracteriza a una ecuación diferencial ordinaria (EDO)? ¿Entiende la diferencia entre una solución general y una solución particular? ¿Puede explicar el rol de las condiciones iniciales?

2. **Ejercicio práctico.** Resuelva la ecuación

$$\frac{dx}{dt} + kx = 0.$$

- Obtenga la solución general.
- Aplique una condición inicial.
- Interprete físicamente la solución.

¿Entiende el comportamiento temporal de la solución?

3. **Ejercicio computacional.** Grafique la solución para distintos valores de k . ¿Entiende cómo cambia la dinámica del sistema?

5.3. Problema homogéneo y problema completo

1. **Pregunta conceptual.** ¿Qué es un problema homogéneo? ¿Qué es un problema completo (no homogéneo)? ¿Entiende la diferencia entre excitación interna y externa?

2. **Ejercicio práctico.** Considere la ecuación

$$\frac{d^2x}{dt^2} + \omega^2x = f(t).$$

- Identifique el problema homogéneo asociado.
- Explique qué representa el término $f(t)$.
- Describa cómo se construye la solución completa.

¿Entiende el principio de superposición?

3. **Ejercicio reflexivo.** Explique qué parte de la solución depende de las condiciones iniciales y cuál de la excitación. ¿Entiende esta separación física?

5.4. Sistemas dinámicos

1. **Pregunta conceptual.** ¿Qué es un sistema dinámico? ¿Entiende la diferencia entre una ecuación de primer orden y un sistema equivalente? ¿Puede explicar qué es el espacio de fases?
2. **Ejercicio práctico.** Transforme la ecuación

$$\frac{d^2x}{dt^2} + \omega^2 x = 0$$

en un sistema de primer orden. Identifique los puntos de equilibrio. ¿Entiende su significado físico?

3. **Ejercicio computacional.** Grafique el retrato de fases del sistema. ¿Entiende qué información aporta este tipo de gráfico?

5.5. Ecuaciones diferenciales parciales

1. **Pregunta conceptual.** ¿Qué diferencia fundamental existe entre una EDO y una EDP? ¿Entiende por qué las EDP aparecen naturalmente en problemas de campos? ¿Puede dar ejemplos físicos asociados a EDP?
2. **Ejercicio práctico.** Clasifique las siguientes ecuaciones:

- Ecuación de Laplace
- Ecuación de Poisson
- Ecuación de onda

¿Reconoce el tipo de fenómeno físico que modelan?

3. **Ejercicio reflexivo.** Identifique las variables independientes y dependientes en cada ecuación. ¿Entiende por qué la solución depende de condiciones iniciales y de frontera?