

MÉTODOS COMPUTACIONALES

CONTENIDO PROGRAMÁTICO

NOMBRE DEL CURSO: Métodos Computacionales

CÓDIGO DEL CURSO: FISI 2028

PROFESOR: Alejandro Segura

UNIDAD ACADÉMICA: Departamento de Física

PRERREQUISITOS: Ecuaciones diferenciales (MATE 2301), Herramientas Computacionales (FISI 2026) o Algorítmica y Programación a Objetos II (ISIS 1205)

I Objetivos

El objetivo principal del curso es desarrollar el pensamiento estructurado para implementar algoritmos numéricos relacionados a:

- Estimar derivadas e integrales.
- Solucionar sistemas de ecuaciones lineales.
- Describir datos con técnicas estadísticas basadas en métodos de Monte Carlo.
- Resolver ecuaciones diferenciales ordinarias y parciales.
- Esquemas reproducibles para el análisis de datos científicos.

II Competencias a desarrollar

Al finalizar el curso, se espera que el estudiante esté en capacidad de:

- Implementar algoritmos numéricos para resolver problemas habituales que aparecen en ciencias e ingeniería.
- Interpretar la información contenida en los datos generados y/o medidos por computador.
- Relacionar la formulación matemática de una idea científica con su implementación computacional.
- Usar adecuadamente técnicas computacionales en un lenguaje de bajo nivel (C++) y otro de alto nivel (Python) para encontrar resultados reproducibles.

III Contenido por semanas

Semana 1 Presentación del curso, consola de Unix, comandos básicos, editores de texto, interprete de Python, git y GitHub.

Lecturas preparatorias: Capítulo 2 (Basic Constructions) del libro de Hans Langtanten.

Semana 2 Repaso de Python, Numpy, Matplotlib, tipos de variables, funciones, ciclos, variables locales y globales, lectura y escritura de datos.

Lecturas preparatorias: Capítulo 4 (Array computing and curve plotting) del libro de Hans Langtanten.

Semana 3. Derivación numérica y raíces de ecuaciones.

Lecturas preparatorias: Capítulos 7.I y 7.II (Differentiation & Searching) del libro de Landau.

Semana 4. Integración numérica e interpolación de Lagrange.

Lecturas preparatorias: Capítulo 6 (Integration) del libro de Landau.

Semana 5. Conceptos básicos de probabilidad y estadística y métodos de Monte Carlo.

Lecturas preparatorias: Capítulo 1 (The basics) del libro de Silvia & Skilling, Capítulo 5 (Monte Carlo Simulations) del libro de Landau.

Semana 6. Estimación bayesiana de parámetros y algoritmo de Metrópolis Hastings.

Lecturas preparatorias: Capítulo 6 (Integration) del libro de Landau.

Semana 7. C++, introducción, sintaxis, compilar/ejecutar, tipos de variables, ciclos y funciones.

Lecturas preparatorias: Capítulo 4 del libro de Deitel.

Semana 8. Condicionales, arreglos, apuntadores, MakeFiles y programación orientación a objetos.

Lecturas preparatorias: Capítulo 3 del libro de Deitel.

Semana 9. Solución de sistemas de ecuaciones lineales y ajuste por mínimos cuadrados.

Lecturas preparatorias: Capítulo 8 (Matrix Equation Solutions) del libro de Landau.

Semana 10. Autovalores y autovectores, análisis de componentes principales (PCA).

Lecturas preparatorias: Secciones 6.3.1 (Principal Component Regression) y 10.2 (Principal Component Analysis) del libro An Introduction to Statistical Learning.

Semana 11. Series de Fourier, transformada de Fourier, señales y filtros.

Lecturas preparatorias: Capítulo 10 (Fourier Analysis) del libro de Landau.

Semana 12. Ecuaciones diferenciales ordinarias, métodos de un paso.

Lecturas preparatorias: Capítulo 9 (ODEs) del libro de Landau.

Semana 13 Ecuaciones diferenciales ordinarias, métodos multipaso y sistemas autónomos.

Lecturas preparatorias: Capítulo 9 (ODEs) del libro de Landau.

Semana 14. Ecuaciones diferenciales parciales. Familias elíptica y parabólica.

Lecturas preparatorias: Capítulo 17 (PDEs) del libro de Landau.

Semana 15. Ecuaciones diferenciales parciales. Familia hiperbólica.

Lecturas preparatorias: Capítulo 17 (PDEs) del libro de Landau.

IV Metodología

La implementación computacional de las ideas teóricas del curso, requiere practica y compromiso. Para alcanzar dichas habilidades, el estudiante debe realizar las lecturas preparatorias antes de cada clase magistral. Posteriormente durante la clase, se hará un resumen de los conceptos teóricos y luego se implementarán como algoritmos computacionales. Adicionalmente, en el laboratorio de métodos computacionales el estudiante podrá profundizar y practicar sobre los temas vistos en la clase magistral.

V Evaluación

El curso se evaluará de la siguiente manera:

- Talleres $\times 4$ (10 %): Entre parciales.
- Primer examen parcial (10 %): Marzo 2.
- Segundo examen parcial (10 %): Abril 6.
- Tercer examen parcial (10 %): Mayo 4.
- Examen final (30 %): \sim Junio 1.

Todos los exámenes, talleres y ejercicios se entregarán a través de SICUA. No se aceptará la entrega de ninguna tarea por fuera de esta plataforma. **Nota: la asistencia a este curso es obligatoria y está sujeta al reglamento de la Universidad y a las últimas reformas académicas.**

VI Bibliografía

Bibliografía principal:

- R.H. Landau y M.J.Páez, *A survey of computational physics*, 2012, Wiley. <https://psrc.aapt.org/items/detail.cfm?ID=11578>
- D.S. Silvia, J. Skilling. *Data Analysis: A Bayesian tutorial*, 2006, Oxford Science Publication.
- H.P. Langtangen, *A Primer on Scientific Programming*, 2009, Springer.
- P. Deitel y H. Deitel, *C++ How to Program*, 2012, Pearson.
- G. James, D. Witten, T. Hastie y R. Tibshirani, *An Introduction to Statistical Learning*, 2013, Springer.
- *Videos del curso Herramientas Computacionales: fundamentos de Unix y Python*. https://www.youtube.com/playlist?list=PLHQtzvthdVM_MGC9dPFKe4hPAwBd_7RJ3

Bibliografía complementaria:

- W. Krauth, *Statistical Mechanics: Algorithms and Computations*, 2006, Oxford University Press.
- M. Lee y D. Henkemans, *C++ programming for the absolute beginner*, 2009, Cengage Learning.
- B. Kernighan y D. Ritchie, *The C programming language*, 1988, Prentice Hall.
- J.V. Guttag, *Introduction to Computation and Programming Using Python*, 2013, The MIT Press.
- *Software Carpentry* <https://software-carpentry.org/>