

HERRAMIENTAS COMPUTACIONALES

CONTENIDO PROGRAMÁTICO

NOMBRE DEL CURSO: Herramientas Computacionales

CÓDIGO DEL CURSO: FISI 2026

UNIDAD ACADÉMICA: Departamento de Física

PRERREQUISITOS: Algorítmica y Programación Orientada por Objetos 1 (ISIS 1204)

I Introducción

Los computadores nos ayudan a organizar, comunicar y procesar información, y hoy en día son esenciales en los mundos de la ciencia, la academia y la técnica. Este curso enseña algunas herramientas computacionales básicas para hacer de los computadores herramientas útiles, poderosas y versátiles. El curso desarrolla habilidades de programación en un lenguaje de alto nivel, por ejemplo Python o Matlab; enseña algunos métodos de análisis numérico; y exhibe algunas herramientas útiles en el análisis de datos.

II Objetivos

Los objetivos del curso son:

- Ofrecer herramientas computacionales básicas útiles en la investigación y la vida académica.
- Introducir rutinas sencillas de análisis numérico.
- Desarrollar habilidades de programación.

III Competencias

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

- Utilizar computadores con sistema operativo tipo UNIX.
- Preparar documentos usando el sistema de composición de textos \LaTeX .
- Implementar en un lenguaje de programación de alto nivel la solución de problemas computacionales sencillos.
- Manipular, analizar y visualizar datos usando un lenguaje de programación de alto nivel.

IV Contenido

Semana 1 [Introducción] Presentación y examen diagnóstico.

Semana 2 [Linux] Introducción a UNIX: filosofía, comandos básicos.

Semana 3 [Linux] Editores de texto, control de procesos, redirección.

Semana 4 [\LaTeX] Lógica de compilación, tipos de documentos, secciones, ecuaciones, tablas y figuras. Manejo de bibliografía con \BibTeX .

Semana 5 [Python] Introducción: filosofía, sintaxis básica, operaciones aritméticas, condicionales.

Semana 6 [Python] Operaciones con cadenas de caracteres, listas y estructuras iterativas.

Semana 7 [Python] Definición de funciones, tipos de variables, recursividad.

Semana 8 [Python] Programación orientada a objetos en Python. Instalación (pip) e importación de módulos. Cuadernos de iPython.

Semana 9 [Python] Introducción a NumPy. Arrays de numpy y operaciones entre arrays.

Semana 10 [Python] Importación de datos. Visualización de datos con matplotlib: plot, scatter, imshow y subplot.

Semana 11 [Python] Análisis numérico: métodos de bisección, método de Newton-Raphson.

Semana 12 [Python] Introducción a SciPy. Ajustes polinomiales y no polinomiales.

Semana 13 [Python] Herramientas estadísticas: funciones estadísticas, histogramas y ejemplos de distribuciones.

Semana 14 [Python] Métodos de Monte Carlo: integración.

Semana 15 [Python] Examen final.

V Metodología

Se hará énfasis en el trabajo individual por fuera del horario de clase. El profesor entregará una lista de recursos que cada estudiante debe preparar. Al inicio de cada clase el profesor responderá preguntas y hará aclaraciones sobre el material de la semana. Luego los estudiantes resolverán los ejercicios de la semana individualmente. Todo el trabajo es individual y las calificaciones se otorgarán por los resultados entregados exclusivamente en horario de clase por SICUA. Bajo ninguna circunstancia se recibirán ejercicios entregados por fuera del tiempo estipulado (hora oficial del final de la clase más 10 minutos). Se recomienda traer a clase una memoria USB si le interesa guardar su trabajo. Se espera del estudiante que evite activamente que su trabajo sea plagiado: cualquier sospecha de copia hará que el caso sea llevado automáticamente a proceso disciplinario, por lo que se aconseja que borre sus archivos del computador que usó cuando no los necesite más. La memoria USB también podrá ser útil si por algún motivo su computador pierde la conexión a internet y necesita cambiar de ordenador para subir su trabajo a la plataforma de SICUA.

VI Calificación del curso

- Examen diagnóstico 20 %
- 3 entregas en clase 60 %
- Examen final 20 %

En doce clases durante el semestre se deberá entregar ejercicios. Estos ejercicios se deben resolver y subir antes de la finalización de la clase a Siciuplus. Entre esos doce ejercicios se elegirán tres al azar que serán calificados, cada uno con un peso del 20 % de la nota final. Con el simple hecho de entregar el examen diagnóstico tendrá la nota máxima en ese ítem.

VII Bibliografía

Bibliografía principal:

- H. P. Langtangen. *A Primer on Scientific Programming with Python*, 2009.
<http://link.springer.com.ezproxy.uniandes.edu.co:8080/book/10.1007%2F978-3-642-18366-9>

Bibliografía complementaria:

- J. V. Guttag. *Introduction to Computation and Programming Using Python*, 2013.
- K. D. Lee. *Python Programming Fundamentals*, 2011.
<http://link.springer.com.ezproxy.uniandes.edu.co:8080/book/10.1007%2F978-1-84996-537-8>
- S. van Vugt. *Beginning the Linux Command Line*, 2009.
<http://link.springer.com.ezproxy.uniandes.edu.co:8080/book/10.1007%2F978-1-4302-1890-6>
- G. Grätzer. *More Math Into L^AT_EX*, 2007.
<http://link.springer.com.ezproxy.uniandes.edu.co:8080/book/10.1007%2F978-0-387-68852-7>