



Plan de Estudios de la Licenciatura en Ciencias de la Tierra

Programa			
Herramientas Computacionales Avanzadas			
Clave	Semestre	Créditos	Campo de conocimiento: Cómputo y Sistemas Informáticos
	4°	6	Etapa de formación: Intermedia
Modalidad	Curso(X) Taller () Lab() Seminario() Otras		Tipo T () P () T/P (X)
Carácter	Obligatorio (X) Optativo () Obligatorio E () Optativo E ()		Horas: 4
Duración	16 semanas		Semana Semestre
			Teóricas: 2 Teóricas: 32
			Prácticas: 2 Prácticas: 32
			Total: 4 Total:64

Seriación	
Ninguna (X)	
Obligatoria ()	
Asignatura antecedente	
Asignatura subsecuente	
Indicativa (X)	
Asignatura antecedente	Herramientas computacionales
Asignatura subsecuente	

Objetivo general:
Desarrollar y aplicar métodos numéricos para la programación de varias funciones.
Objetivos particulares:
Manipular bases de datos por diferentes métodos numéricos.
Programar en dispositivos externos de fácil acceso.
Diseñar circuitos con dispositivos para el almacenaje de datos e información a usar en posteriores análisis.
Comparar la versatilidad de la programación con otro software disponible.

Índice temático			
	Tema	Horas	
		Semestre	
		Teóricas	Prácticas
1	Métodos numéricos	8	8
2	Técnicas de cálculo numérico en varias variables	8	8
3	Interpolación de funciones	8	8
4	Comunicación y manipulación de dispositivos	8	8
Subtotal		32	32



Total	64
--------------	-----------

Contenidos temáticos	
Temas	Subtemas
1	Métodos numéricos 1.1 Diferenciación numérica de una variable y control de error. 1.2 Integración numérica en una variable y criterios de convergencia. 1.3 Diferencias finitas y su relación con el cálculo numérico y las ecuaciones diferenciales.
2	Técnicas de cálculo numérico en varias variables 2.1 Programación de la derivada direccional. 2.2 Integración en varias variables. 2.3 Integración de Crank-Nicholson. 2.4 Integración de ecuaciones en derivadas parciales de segundo orden de Tipo parabólico, elíptico e hiperbólico.
3	Interpolación de funciones 3.1 Interpolación polinomial y splines. 3.2 Interpolación de funciones escalares para realizar ajustes topográficos. 3.3 Integración numérica a lo largo de una línea y del flujo de un campo vectorial.
4	Comunicación y manipulación de dispositivos 4.1 Lectura de datos en tiempo real a través de puertos de comunicación. 4.2 Programación en dispositivos externos como Arduinos, RaspberryPi, BeagleBone Black, y similares. 4.3 Construcción primaria de circuitos con dispositivos de entrada/salidad de datos e información. 4.4 Interconexión entre diferentes interfaces de programas y datos.

Estrategias didácticas
Trabajo en equipo
Aprendizaje basado en problemas
Exposición oral

Evaluación del aprendizaje
Trabajos y tareas
Exámenes parciales
Examen final
Elaboración de ensayos

Perfil profesiográfico del docente	
Título o grado	Matemático(a), Actuario(a), Lic. Ciencias de la Computación, Ing. En Sistemas Computacionales. Nivel mínimo de Maestría.
Experiencia docente	Experiencia docente de al menos dos años en el manejo de los distintos lenguajes de programación y software libre. Haber tomado el curso de formación docente impartido por la ENCIT.
Otras	Manejo de diferentes plataformas y programas de software libre

características	
-----------------	--

Bibliografía básica

Angela B. Shiflet y George W. Shiflet. (2013) Introduction to Computational Science: Modeling and Simulation for the Sciences – Second Edition. Princeton University Press; Edición: 2 (30 de marzo de 2013)

Press WH, Teukolsky SA, Vetterling WT, Flannery BP. (2007) Numerical recipes, the art of scientific computing. 3rd ed. Cambridge (UK): Cambridge University Press; 2007.

Newman M. (2012) Computational Physics. CreateSpace Independent Publishing Platform; 2012.

Guy Simpson. (2017) Practical Finite Element Modeling in Earth Science using Matlab. Wiley-Blackwell; Edición: 1 (17 de abril de 2017)

Mesografía

<https://github.com/>

<http://www.scipy.org>

<http://jupyter.org>

<https://julialang.org/>

<http://matplotlib.org>

Bibliografía complementaria

John M. Stewart.(2014) Python for Scientists. Cambridge University Press; Edición: 1 (29 de agosto de 2014)

Downey AB, Elkner J, Meyers C.(2002) How to think like a computer scientist: learning with Python. USA: Green Tea Press; 2002.

Mesografía

<http://www.scipy.org>

<http://www.python.org>

<http://www.sagemath.org/>