

Este programa esta en proceso de aprobación por el Consejo de Facultad (última actualización Tue, 09 Feb 2016 13:27:12 -050). Úselo solamente como fuente de información preliminar. Una versión previa del curso puede encontrarse en el enlace:

<http://astronomia-udea.co/principal/Curriculo/planes.php>

Allí se publicará también la versión definitiva de este semestre una vez este aprobado.

Este curso esta en edición y no es una versión distribuible. Esta disponible para edición en:

<http://astronomia-udea.co/principal/Curriculo/links/d45fba.html>.

FORMATO DE MICROCURRICULO O PLAN DE ASIGNATURA

1. IDENTIFICACIÓN GENERAL			
Facultad	Facultad de Ciencias Exactas y Naturales		
Instituto	Instituto de Física		
Programa(s) Académicos	Astronomía, Física		
Área Académica	Computación		
Ciclo	Fundamentación		
Tipo de Curso	Básico		
Profesores Responsables	Nicolás Gómez, Luis F. Quiroga, Cesar Alfredo Uribe, Sheryl Avendaño		
Asistencia	Obligatoria		
2. IDENTIFICACIÓN ESPECÍFICA			
Semestre	2014-2		
Nombre de la Asignatura	Fundamentación en Computación		
Código	0302150		
Semestre en el plan	1		
Número de Créditos	2		
Horas Semestrales	HDD:64	HDA:0	TI:32
Semanas	16		
Intensidad Semanal	Teórico: 0	Práctico: 0	Teórico-Práctico: 4
H (Habilitable)	Si		
V (Validable)	Si		
C (Clasificable)	No		
Prerrequisitos	Ninguno		
Correquisitos	Ninguno		
Sede en la que se dicta	Ciudad Universitaria Medellín		
3. DATOS DE LOS PROFESORES QUE ELABORAN EL PLAN DE ASIGNATURA			
Nombres y Apellidos	Mario Alexander Sucerquia, Jorge Zuluaga, Esteban Silva Villa		

Correo Electrónico	malsuar@gmail.com , zuluagajorge@gmail.com , esteban.silvav@udea.edu.co
--------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

4. DESCRIPCIÓN

Se exponen en este curso dos tópicos generales: a) una introducción general al sistema operativo Linux, la estructura de dicho sistema y las herramientas básicas para su manejo (consola de comandos, conexiones remotas), y b) una introducción al lenguaje de programación compilado Python con un enfoque general, partiendo de la algoritmia y finalizando con la codificación.

Se aborda además la temática de la representación de los datos, introduciendo algunas herramientas, librerías de graficación y análisis de acceso libre (Matplotlib y Scipy).

La gestión de los datos: Procesamiento, representación su presentación, son los tópicos fundamentales que constituyen este curso.

5. JUSTIFICACIÓN

El quehacer académico y científico gira alrededor de los datos: su obtención, manipulación, almacenamiento, representación y presentación de la información extraída a partir de ellos. Para esta labor existen y se desarrollan constantemente herramientas computacionales que facilitan estas operaciones y que el científico en formación debe conocer y manipular adecuadamente. Entre estas herramientas se pueden enumerar, los lenguajes de programación, los manipuladores de archivos y los programas y librerías de programación.

Manejar adecuadamente las herramientas computacionales disponibles que le permiten al científico solucionar problemas numéricos mediante procesos automatizados, economizando tiempo de labor humana. Le permite verificar modelos teóricos mediante simulaciones y finalmente realizar una gestión completa de la información. El estudiante en formación debe conocer cuáles son las posibilidades que le ofrece el computador, al igual que sus limitaciones.

La programación del computador permite además de lo mencionado, aprender a pensar de modo lógico y algorítmico, habilidad imprescindible para el científico en cualquiera de sus quehaceres, ya sean computacionales o no.

6. OBJETIVOS

Objetivo General:

Dotar al estudiante de las herramientas computacionales básicas necesarias para la manipulación, procesamiento, representación de datos y presentación de la información, a través del desarrollo de un pensamiento algorítmico y la codificación de instrucciones usando lenguajes de programación, graficación y entornos de desarrollo integrados, para facilitar el desarrollo de sus tareas académicas y científicas futuras.

Objetivos Específicos:

Al terminar el semestre el estudiante podrá:

Objetivos Conceptuales:

Proponer algoritmos para solucionar problemas simples.

Identificar los procesos de manipulación de datos a nivel de hardware y software bajo los cuales opera el computador.

Comprender la importancia de la computación en la solución de problemas científicos desde el punto de vista de la automatización, la manipulación y la visualización de la información.

Reconocer la existencia de una amplia gama de herramientas computacionales útiles en el quehacer académico y científico.

Objetivos Actitudinales:

Desarrollar la capacidad de pensar algorítmicamente en la solución de problemas.

Fomentar la capacidad de programar empleando diferentes lenguajes de programación.

Crear en el alumno la necesidad de emplear el computador como herramienta para resolver problemas de ciencias en general.

Potencializar las habilidades de búsqueda y gestión bibliográfica de forma crítica.

Objetivos Procedimentales:

Familiarizarse con la elaboración de algoritmos, diagramas de flujo y pseudocódigos como herramientas para la solución de problemas computacionales.

Emplear diferentes lenguajes de programación para codificar instrucciones que le permitan solucionar diferentes problemas de tipo numérico.

Usar herramientas de graficación de gran calidad para la representación de datos.

Presentar informes usando altos estándares tipográficos con herramientas computacionales libres.

7. CONTENIDOS

Contenido Resumido

- 1-Introducción a el sistema operativo Linux
- 2-Introducción a la algorítmia
- 3-Introducción al lenguaje Python

Unidades Detalladas

Unidad 1. Introducción a el sistema operativo Linux (2 semanas)

Contenidos conceptuales:

Sistema operativo Linux, manipulación de archivos, manejo de la Shell, comandos básicos, uso de editores de texto plano. Representación decimal, binaria y hexadecimal, representación de caracteres, operaciones con enteros y de punto flotante, uso de paréntesis, incrementos y decrementos, comparación de números, toma de decisiones.

Contenidos procedimentales:

Reconocer las partes del sistema operativo Linux a través del uso de la consola de comando, la conexión remota y la manipulación de archivos.

Realizar actividades que conlleven el aprendizaje de las herramientas básicas del sistema operativo.

Contenidos actitudinales:

Reconocer la importancia del uso de el sistema operativo Linux.

Diferenciar las partes que componen el sistema operativo Linux y como usarlas.

Unidad 2. Introducción a la algorítmia (3 semanas)

Contenidos conceptuales:

Elementos de pseudo-código, diagramas de flujo, variables, rutinas y condicionales.

Contenidos procedimentales:

Reconocer las diferentes partes de los metodos para generar algoritmos computacionales.

Diagramar los algoritmos a través de diagramas de flujos conceptuales.

Aprender los diferentes tipos de variables, rutinas y condicionales necesarios para la programación.

Contenidos actitudinales:

Reconocer la importancia del uso de el los pseudo-códigos para la elaboración de programas computacionales.

Diferenciar las partes que componen la algorítmia.

Unidad 3. Introducción al lenguaje Python (11 semanas)

Contenidos conceptuales:

Elementos principales del código Python, variables, condicionales, listas y vectores, iteraciones, subrutinas, escritura y lectura de archivos, Numpy, Matplotlib y Scipy/Panda

Contenidos procedimentales:

Reconocer los diferentes aspectos del lenguaje de programación Python.

Realizar diferentes tipos de códigos para la solución de problemas en diversos ambitos.

Contenidos actitudinales:

Reconocer la importancia del uso de el lenguaje de programación Python

Reconocer la importancia de la programación para la solución de problemas en las áreas de las ciencias exactas.

8. ESTRATEGIAS METODOLÓGICAS

El curso será desarrollado a través de una metodología que vincula los siguientes aspectos:

La clase-taller: Esta materia se fundamenta en el aprendizaje mediante el ejemplo: en la sesión de clase el profesor expone el funcionamiento básico de las instrucciones y se realizan ejemplos de su uso. Posteriormente reta al estudiante a realizar actividades análogas, empleando diversas áreas del conocimiento científico.

Actividades independientes y asistidas: Se propone una amplia gama de actividades independientes con periodicidad semanal donde el estudiante debe aplicar los conceptos tratados en clase en problemas heterogéneos. Parte de la realización de estas actividades será asesorada por el profesor durante un segmento de la clase.

9. EVALUACIÓN

Cuatro (4) exámenes parciales 3 del 15% c/u y 1 del 10% Total 55 %

Sies (6) tareas y diez (10) quises del 2% c/u Total 30 %

Un (1) proyecto final del 15% Total 15 %

Total 100%

10. BIBLIOGRAFÍA

Learning Python, Mark Lutz.

Introducción a la programación con Python, Andrés Marzal.

<http://code-reference.com>

Manuales y documentación de referencia librerías python online.

Learning Python, Mark Lutz.

Introducción a la programación con Python, Andrés Marzal.

<http://code-reference.com>

Manuales y documentación de referencia librerías python online.

Última actualización: Tue, 09 Feb 2016 13:30:14 -0500

Versión legal: La versión legal de este documento reposa en la Biblioteca de la Universidad de Antioquia y esta firmada por el Decano y el Director de Instituto.

Firma Autorizada Facultad Versión Electrónica: (No autorizado. Este documento es solo un borrador.)