

Programación Científica II**IDENTIFICACIÓN DE LA ACTIVIDAD CURRICULAR**

Nombre de la Actividad	:	COMPUTACIÓN CIENTÍFICA II					
Código	:	DIN-124					
Semestre lectivo	:	II					
Horas	:	Presencial:	72	Autónomas:	108	Total:	180
Créditos SCT	:	6					
Duración	:	Semestral:	X	Anual:			
Modalidad	:	Presencial:	X	Semi-presencial:			
Área de Formación		GENERAL					
Requisito		COMPUTACIÓN CIENTÍFICA I					

DESCRIPCIÓN Y CARACTERIZACIÓN DE LA ACTIVIDAD CURRICULAR

Esta asignatura es la continuación de Computación Científica I para fortalecer las competencias asociadas a la resolución de problemas matemáticos mediante la computación. Al finalizar el curso, el estudiante será capaz de utilizar estructuras de datos, manejo de datos y computación paralela, para resolver problemas de cómputo científico.

COMPETENCIAS DEL PERFIL DE GRADO ASOCIADAS A LA ACTIVIDAD CURRICULAR.

COMPETENCIA ESPECÍFICA	SUBCOMPETENCIA
2. Proponer sistemas tecnológicos o soluciones innovadoras, basados en la toma de datos en terreno, la experimentación y/o la modelación para la formulación; desde la Ingeniería, de soluciones eficientes y sostenibles.	2.1 Diseñar soluciones tecnológicas eficientes y sostenibles basadas en el análisis del estado del arte de un problema.

COMPETENCIAS GENÉRICAS	SUBCOMPETENCIA
5. Realizar investigaciones que contribuyan al desarrollo del conocimiento científico y aplicado, en el contexto propio de su proceso formativo.	5.1 Analizar resultados de investigación en Ingeniería actualizados relacionados con las líneas de Sistemas Inteligentes y Reconocimiento de Patrones, Sistemas Silvoagrícolas Digitales, o Bioingeniería en Ambientes Naturales y Modificados, con el fin de construir el estado del arte de un problema científico y tecnológico.

RESULTADOS DE APRENDIZAJE.

RESULTADOS DE APRENDIZAJES (R.A.)
1. Resolver problemas de cómputo científico utilizando estructuras de Datos Lineales y no Lineales adecuadas para el problema, en base a consideraciones de eficiencia de tiempos de acceso, utilización de recursos, tipo de implementación y complejidad.
2. Utilizar técnicas y herramientas de visualización de datos para el análisis de un problema de cómputo científico.
3. Resolver eficientemente problemas de cómputo científico, usando técnicas de computación paralela junto a la plataforma de hardware adecuada.

UNIDADES DE APRENDIZAJE Y EJES TEMÁTICOS

R.A.	UNIDAD	EJE(S) TEMÁTICO(S)
1	Estructuras de Datos y Manejo de Arboles y Grafos	Recursividad Tipos de Datos Abstractos Lineales y no Lineales (Lista enlazada, Pila, Cola, Arboles, Grafos). Métodos de recorrido de arboles, Búsqueda Binaria, Heaps. Composición de un grafo, Matriz de Adyacencia, Recorrido y Búsqueda de caminos mínimos.
2	Manejo y Visualización de Datos	Manejo de archivos estructurados y no estructurados (exploración y manipulación de datos). Visualización de los Datos. Clustering y Reducción de Dimensionalidad.
3	Computación Paralela	Introducción a Sistemas de Hardware Paralelo y Distribuido. Programación multi-hilo (identificación de tareas independientes, memoria compartida y privada, regiones críticas, sincronizaciones).

ESTRATEGIAS DE ENSEÑANZA Y APRENDIZAJE

<p>La metodología de trabajo para el desarrollo de la actividad curricular, se basa en un enfoque de resolución de ejercicios y problemas en el aula; esto implica entregar un rol protagónico al estudiante que es entendido como eje y centro de acción, y quién a través de su participación activa y con orientaciones y lineamientos que le entrega el docente va construyendo su propio aprendizaje. Para lograr este objetivo, las distintas clases consideran una serie de estrategias metodológicas, previamente seleccionadas por el docente:</p> <ul style="list-style-type: none">- Método expositivo-participativo.- Aprendizaje en base a resolución de ejercicios y problemas en el aula.- Aprendizaje en base a resolución de ejercicios y problemas con uso de software en laboratorio de computación.

PROCEDIMIENTOS DE EVALUACION DE APRENDIZAJES.

Resultado de Aprendizajes	Indicadores de Evaluación	Estrategia de Enseñanza y Aprendizaje	Técnica/ Instrumento Evaluativo	Ponderación (%)
1.- Resolver problemas de cómputo científico utilizando estructuras de Datos Lineales y no Lineales adecuadas para el problema, en base a consideraciones de eficiencia de tiempos de acceso, utilización de recursos, tipo de implementación y complejidad.	Uso de estructuras de datos lineales y no lineales. Implementación correcta de un algoritmo recursivo. Correcta implementación y manejo de un grafo.	Clase expositiva. Resolución de problemas de programación en clase. Talleres de Laboratorio.	Tarea de Laboratorio / Pauta. Exposición de la tarea / Pauta.	20%
2. Utilizar técnicas y herramientas de visualización de datos para el análisis de un problema de cómputo científico.	Manejo correcto de archivos estructurados y no estructurados. Uso adecuado de herramientas de visualización de datos para un problema específico. Aplicación correcta de técnicas de análisis de datos.	Clase expositiva. Resolución de problemas de programación en clase. Talleres de Laboratorio.	Tarea de Laboratorio / Pauta. Exposición de la tarea / Pauta.	20%
3. Resolver eficientemente problemas de cómputo científico, usando técnicas de computación paralela junto a la plataforma de hardware adecuada.	Implementación eficiente de algoritmos multi-hilo. Resolución de un problema de cómputo científico mediante el uso eficiente de herramientas de programación.	Clase expositiva. Resolución de problemas de programación en clase. Talleres de Laboratorio.	Tarea de Laboratorio / Pauta. Exposición de la tarea / Pauta. Desarrollo de Proyecto Integrador / Pauta.	60%

RECURSOS DE INFRAESTRUCTURA

Sala de clases, Servidores de cómputo LITRP, Proyector, Telones, Pizarras amplias, Compiladores e Intérpretes, Biblioteca, Sistemas LMS-UCM.

RECURSOS BIBLIOGRÁFICOS

	Autor, año; Título del trabajo, año edición, lugar de publicación, editorial. (De acuerdo a tipo de material consultado y de acuerdo a la normativa APA 6° Edición)
BÁSICA OBLIGATORIA	<ul style="list-style-type: none"> Joakim Sundnes (2020): Introduction to Scientific Programming with Python. Simula SpringerBriefs on Computing. https://doi.org/10.1007/978-3-030-50356-7 Chazallet, Sébastien (2016): Python 3 : Los fundamentos del lenguaje. Barcelona : Ediciones ENI. Luis Joyanes Aguilar (2017): FUNDAMENTOS DE PROGRAMACIÓN. Algoritmos, estructura de datos y objetos. Cuarta edición. McGraw Hill. Tim Grobmann, Mario Dobler, Data Visualization with Python, O'Reilly, 2019. Qingkai Kong, Timmy Siau, Alexandre Bayen. Python Programming And Numerical Methods: A Guide For Engineers And Scientists. Academic Press, 2020.
COMPLEMENTARIA	<ul style="list-style-type: none"> Joel Grus, Data Science from Scratch, First Principles with Python, O'Reilly, 2015. Jake VanderPlas, Python Data Science Handbook Essential Tools for Working with Data, O'Reilly, 2016. Giancarlo Zaccone, Python Parallel Programming Cookbook, Packt 2019.

OTROS RECURSOS

Nombre Recurso	Tipo de Recurso
Libros y revistas	Digital
Sitios Web	Digital
Sitio Web: Documentación Oficial de Python.	Digital, URL: https://docs.python.org
Sitio Web: Python Programming and Numerical Methods.	Digital, URL: https://pythonnumericalmethods.berkeley.edu/notebooks/Index.html
Repositorio de Código.	Digital, URL: https://github.com/PacktPublishing/Applying-Math-with-Python