



1. Identificación del curso

Fundamentos de programación							
Programa educativo				Departamento de adscripción			
Ingeniería en Computación				Departamento de Ingenierías			
Área de formación				Tipo de Unidad de Aprendizaje			
Básica común obligatoria				Curso - Taller			
Carga horaria					Créditos		Clave
Teoría	40	Práctica	40	Total	80	8	I5288
Modalidad de Enseñanza - Aprendizaje				Prerrequisito			
Mixta				Ninguno			
Academia				Profesor responsable			
Ciencias computacionales				Claudia Islas Torres			
Elaboró / Modificó				Fecha de elaboración / modificación			
María Obdulia González Fernández / Héctor González Sánchez / Claudia Islas Torres				10 de mayo de 2024			

2. Competencias que abonan al perfil de egreso

Transversal	Disciplinar	Profesional
Posee habilidades de aprendizaje autogestivo que le permiten incrementar sus conocimientos en distintas áreas de interés.	Demuestra conocimientos y habilidades en la aplicación de procedimientos algorítmicos en el uso de las tecnologías de la información para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos.	Diseñar sistemas de software y de información, implementando arquitecturas, infraestructuras y características de seguridad, para dar solución a problemáticas reales.

3. Saberes previos

Matemáticas; Informática y Comunicación

4. Presentación de la unidad de aprendizaje

En esta unidad de aprendizaje se establecen las bases de la programación, las estructuras de datos y la programación en el lenguaje C. Asimismo, es importante destacar que las bases adquiridas en dicha unidad de aprendizaje son necesarias para implementar la parte práctica de diversas asignaturas a lo largo de la carrera.

En esta materia se establecen los principios generales del proceso de abstracción requerido para analizar y comprender una situación o un problema, para entonces elaborar una descripción amplia en términos precisos y libres de ambigüedad.

Para que una descripción así pueda ser procesada por una computadora, inicialmente se expresa en un lenguaje definido mediante reglas formales de estructuración (pseudocódigo), que deberá traducirse a algún lenguaje de programación específico.

Esta asignatura aporta, al perfil del ingeniero la capacidad para desarrollar pensamiento lógico, identificar el proceso de creación de un programa y desarrollo de algoritmos para resolver problemas. Es por ello que, el estudiante requiere de pensamiento lógico, identificar el proceso de creación de un programa y el desarrollo de algoritmos.

5. Objetivo de aprendizaje

Analizar problemas para crear algoritmos, pseudocódigos e implementarlos en un lenguaje de programación.

6. Competencia general de la unidad de aprendizaje

CG.CI.170 Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.

7. Habilidades, valores y actitudes

Capacidad para gestionar y formular proyectos. Comprensión lectora y análisis crítico al buscar información procedente de diversas fuentes sobre las temáticas de la unidad de aprendizaje. Trabajo en equipos multi, inter y transdisciplinarios.



8. Elementos de competencia

Bloque No. I: Introducción a la programación		
Sub-competencia	El estudiante comprende los conceptos principales asociados a la programación	
Cognitivos (Contenido)	• Conceptos asociados a la programación (tipos de software, programas, sistemas) • Concepto de programación • Paradigmas de programación • Tipos de lenguajes de programación (comparación de lenguajes) • Entornos de desarrollo integrado (IDE), interfaz gráfica de usuario (GUI) y los compiladores	
Procedimentales	Analiza y compara los diferentes tipos de lenguajes de programación y los traductores	
Estrategias didácticas	Aula invertida, ABP y aprendizaje basado en proyectos	
Criterios de desempeño	Producto esperado	Sesiones estimadas
El estudiante diseña algoritmos sencillos para resolver problemas de la vida cotidiana	Organizadores gráficos sobre la temática, reportes de información sobre la temática	6 sesiones
Área de conocimiento	6.1 Algoritmia	

Bloque No. II: Herramientas para la resolución de problemas de programación		
Sub-competencia	El estudiante comprende el proceso de construcción de un algoritmo y su relación con los lenguajes de programación	
Cognitivos (Contenido)	• Fases de la resolución de un problema (Ciclo de vida del desarrollo de software clásico) • El algoritmo ○ Los diagramas de flujo ○ El pseudocódigo • El proceso de compilación ○ Aplicación de algoritmos en diferentes lenguajes • Errores típicos de la programación	
Procedimentales	Desarrolla algoritmos mediante diagramas de flujo y pseudocódigo, reconoce el proceso para programar un algoritmo en un lenguaje de programación determinado	
Estrategias didácticas	Aula invertida, ABP y aprendizaje basado en proyectos	
Criterios de desempeño	Producto esperado	Sesiones estimadas
El estudiante diseña algoritmos sencillos para resolver problemas de la vida cotidiana	Diagramas de flujo, pseudocódigos y primeras prácticas lenguaje C	12 sesiones
Área de conocimiento	6.1 Algoritmia	

Bloque No. III: Fundamentos de la programación	
Sub-competencia	El estudiante es capaz de comprender los conceptos de variables y estructuras secuenciales y de control
Cognitivos (Contenido)	



- Variables y constantes y tipos de datos básicos.
- Sentencias e instrucciones secuenciales: entrada y salida de datos
- Tipos de operadores
 - Operadores de Asignación
 - Operadores aritméticos
 - Operadores lógicos
 - Operadores relacionales o de comparación
 - Operadores de dirección
- Estructuras de control
 - Estructuras de control selectivas
 - Estructuras de control iterativas
- Palabras reservadas

Procedimentales

Resuelve problemas desarrollando algoritmos secuenciales de cualquier tipo. Representados en diagramas de flujo y pseudocódigos. Comprende el funcionamiento de las estructuras de control tanto condicionales como cíclicas.

Estrategias didácticas

Aula invertida, ABP y aprendizaje basado en proyectos

Criterios de desempeño	Producto esperado	Sesiones estimadas
Genera pseudocódigo de manera lógica y ordenada. Demuestra mediante pruebas de escritorio el correcto funcionamiento de un algoritmo. Argumenta la pertinencia de sus algoritmos. Transfiere un algoritmo a un pseudocódigo y un diagrama de flujo	Tareas grupales como: mapas conceptuales, cuadros comparativos de las estructuras de control. Pseudocódigo y desarrollo de código en un lenguaje de programación	12 sesiones
Área de conocimiento	6. Programación e Ingeniería de software	

Bloque No. IV: Prácticas de programación

Sub-competencia El estudiante aplica sus aprendizajes sobre datos y estructuras secuenciales y de control

Cognitivos (Contenido)

- Estructuras de programación

Procedimentales

Resuelve problemas aplicando sus conocimientos sobre algoritmos. Representados en diagramas de flujo y pseudocódigos. Comprende el funcionamiento de las estructuras de control tanto condicionales como cíclicas.

Estrategias didácticas

Aula invertida, ABP y aprendizaje basado en proyectos

Criterios de desempeño	Producto esperado	Sesiones estimadas
Genera pseudocódigo de manera lógica y ordenada. Transfiere un algoritmo a un pseudocódigo y un diagrama de flujo Presenta el código desarrollado lenguaje C y Python	Tareas grupales como: mapas conceptuales, cuadros comparativos de las estructuras de control. Pseudocódigo y desarrollo de código en un lenguaje de programación	12 sesiones
Área de conocimiento	6. Programación e Ingeniería de software	

9. Recursos requeridos

Computadora, bibliográfica recomendada, videos y plataforma educativa



10. Evaluación y acreditación de la unidad de aprendizaje

Actividades de documentación	15%
Tareas y prácticas en laboratorio	50%
Evaluaciones parciales	30%
Participación en clase	5%

11. Referencias (APA)

Básica

Sznajdleder P. (2011). Algoritmo a fondo con implementaciones en C y Java. Alfa Omega, 1.^a edición México. 553
Lee R. (2007). Introducción al diseño y análisis de algoritmos. MC Graw Hil 1.^a edición, México, pp736
López G. (2009). Análisis y Diseño de Algoritmos, Alfa Omega, 1.^a edición, México.
Joyanes L. (2008). Fundamentos de programación, 4ta edición, México

Complementaria

Pinales Delgado, F. J. y Velázquez Amador C. E. (2014). Problemario de algoritmos resueltos
Con diagramas de flujo y pseudocódigo. Universidad Autónoma de Aguascalientes.
Camacho, S. (1996). Análisis de algoritmos. México: U.N.A.M., E.N.E.P. Acatlán

Sitios web

C++ Tutorial. (n.d.). Retrieved December 15, 2021, from <https://www.w3schools.com/cpp/default.asp>

12. Campo de aplicación profesional

Demuestra conocimientos y habilidades en la aplicación de procedimientos algorítmicos en el uso de las tecnologías de la información para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos

13. Perfil docente

El docente de esta materia deberá ser un profesionalista con formación en las áreas de la computación, comunicaciones o informática; capaz de motivar a la investigación y creación de conocimientos con habilidades para transmitir sus conocimientos y enseñar de forma interactiva, propiciando en los alumnos el auto-aprendizaje.



CENTRO UNIVERSITARIO DE LOS ALTOS
DIVISIÓN DE CIENCIAS AGROPECUARIAS E INGENIERÍAS
DEPARTAMENTO DE INGENIERÍAS

Dr. Alejandro Pérez Larios
Jefe de Departamento de ingenierías

Mtro. Fernando Cornejo Gutiérrez
Presidente de academia