



UNIVERSIDAD AUTÓNOMA DE SINALOA
FACULTAD DE INGENIERÍA MOCHIS
LICENCIATURA EN INGENIERÍA EN PROCESOS INDUSTRIALES



PROGRAMA DE ESTUDIO

1. DATOS DE IDENTIFICACIÓN			
UNIDAD DE APRENDIZAJE O MÓDULO:	PROGRAMACION DE COMPUTADORAS		
Clave:	2440		
Ubicación:	SEMESTRE 4	Área: CURSOS COMPLEMENTARIOS	
Horas y créditos:	Teóricas: 32	Prácticas: 48	Estudio Independiente: 48
	Total de horas: 128		Créditos: 6
Competencia(s) del perfil de egreso al que aporta:	Desarrollar habilidades para analizar problemas complejos y proponer soluciones mediante algoritmos y programación. Fomentar el pensamiento lógico y crítico, permitiendo a los estudiantes evaluar diferentes enfoques para resolver un problema.		
Unidades de aprendizaje relacionadas:	MÉTODOS NUMÉRICOS		
Responsable(s) de elaborar el programa:	Dr. Horacio Edgardo Garrafa Gálvez Ing. Salvador Barajas Rodríguez Dr. Carlos Alberto Peñuelas Gámez		Fecha: 24/09/2024
Responsable(s) de actualizar el programa:			Fecha:
2. PROPÓSITO			
Formar ingenieros competentes que puedan utilizar la programación como una herramienta clave en la mejora de procesos y la toma de decisiones en el ámbito industrial.			
3. SABERES			
Teóricos:	- Comprensión de conceptos básicos como variables, tipos de datos, operadores, estructuras de control (if, switch, bucles), y funciones. - Desarrollo y análisis de algoritmos para resolver problemas específicos, incluyendo técnicas de diseño de algoritmos y complejidad computacional. - Aprendizaje de los principios de la POO, como clases, objetos, herencia, polimorfismo y encapsulamiento, que permiten desarrollar software más modular y reutilizable. - Conocimiento de diferentes enfoques y metodologías para el desarrollo de software, como Agile, Scrum y Waterfall.		



UNIVERSIDAD AUTÓNOMA DE SINALOA
FACULTAD DE INGENIERÍA MOCHIS
LICENCIATURA EN INGENIERÍA EN PROCESOS INDUSTRIALES



PROGRAMA DE ESTUDIO

	- Comprensión de cómo se aplica la programación para optimizar procesos industriales, análisis de datos, simulaciones y automatización de tareas.
Prácticos:	- Habilidad para escribir código en uno o más lenguajes de programación (como Python, Java, C++, etc.), aplicando la sintaxis y las convenciones del lenguaje. - Capacidad para desarrollar y codificar algoritmos que resuelvan problemas específicos, implementando lógica y secuencias de instrucciones. - Aplicación práctica de diferentes estructuras de datos para gestionar y organizar información de manera eficiente. - Habilidades para colaborar con otros estudiantes en proyectos de programación, utilizando herramientas de control de versiones (como Git) y metodologías de desarrollo ágil. - Experiencia en la aplicación de la programación para abordar problemas reales en el contexto industrial, desarrollando soluciones prácticas.
Actitudinales:	- Compromiso con la entrega puntual de trabajos y proyectos, así como la calidad del código y la documentación presentada. - Actitud positiva hacia el trabajo en equipo, fomentando la comunicación y la cooperación con compañeros en proyectos grupales. - Habilidad para enfrentar y superar desafíos, como errores en el código o dificultades en la implementación de algoritmos, manteniendo una actitud positiva. - Actitud de servicio hacia el usuario final, entendiendo la importancia de crear soluciones que respondan a sus necesidades y expectativas. - Compromiso con el aprendizaje continuo y la práctica constante, reconociendo que la programación requiere dedicación y esfuerzo.

4. CONTENIDOS

UNIDAD I. INTRODUCCIÓN A LAS COMPUTADORAS Y A LOS LENGUAJES DE PROGRAMACION

- 1.1. Reseña histórica de la computadora
 - 1.2. Conceptos y Definiciones
 - 1.2.1. Computadora y sus componentes
 - 1.2.2. Tipos de software
 - 1.3. Lenguajes de programación
 - 1.3.1. Tipos de lenguajes de programación
 - 1.3.2. Componentes (Editor, Intérprete, Compilador, Depurador)
- Práctica de Laboratorio #1: "Operaciones básicas, hoja de cálculo de Excel" Práctica de Laboratorio #2: "Fórmulas, gráficos y funciones en Excel" Práctica de Laboratorio #3: "Conocer el ambiente de trabajo del Lenguaje de Programación"

UNIDAD II. ALGORITMOS

- 2.1. Conceptos y Definiciones
 - 2.1.1. Algoritmo
 - 2.1.2. Representación de un algoritmo



UNIVERSIDAD AUTÓNOMA DE SINALOA
FACULTAD DE INGENIERÍA MOCHIS
LICENCIATURA EN INGENIERÍA EN PROCESOS INDUSTRIALES



PROGRAMA DE ESTUDIO

- 2.2. Metodología para el diseño de algoritmos
 - 2.2.1. Fases para la resolución de problemas en computadora
 - 2.2.2. Prueba de escritorio
- 2.3. Tipos de Programación Práctica de Laboratorio #4: “Calcular la raíz cuadrada de N número, con la fórmula de Newton, utilizando Excel”

UNIDAD III. FUNDAMENTOS DEL LENGUAJE

- 3.1. Estructura de un programa
 - 3.1.1. Definición de programa
 - 3.1.2. Componentes de un programa
- 3.2. Variables y Tipos de datos
- 3.3. Expresiones y operadores (Aritméticos, Lógicos y Relacionales)
- 3.4. Interfaz Gráfica de Usuario (GUI)
- 3.4. Estructuras de control:
 - 3.4.1. Secuencial, condicional e iterativas
- 3.5. Funciones
 - 3.5.1. Funciones básicas del Lenguaje
 - 3.5.2. Creación de funciones propias
- 3.6. Gráficos Práctica de Laboratorio #5: “Operaciones Básicas, con interfaz gráfica” Práctica de Laboratorio #6: “Calculadora Básica, con interfaz gráfica” Práctica de Laboratorio #7: “Calcular el área de un triángulo, con interfaz gráfica” Práctica de Laboratorio #8: “Tabular y graficar una función, con interfaz gráfica” Práctica de Laboratorio #9: “Uso de estructuras de Control: Condicional y repetitivas” Generar Script para: • Resolver ecuaciones de 2do. Grado (If) • Calcular el Factorial de N número (while y for)

UNIDAD IV. ARREGLOS

- 4.1. Definición y tipos de arreglos
- 4.2. Funciones con arreglos
- 4.3. Operaciones básicas con arreglos Práctica de Laboratorio #10: “Operaciones Básicas con matrices” Práctica de Laboratorio #11: “Uso de funciones para manejar arreglos”

UNIDAD V. ARCHIVOS Y PUERTOS

- 5.1. Definición de archivo y tipos de archivos
- 5.2. Operaciones básicas con archivos
- 5.3. Entrada y salida de datos mediante archivos
- 5.4. Puertos
 - 5.4.1. Definición de puerto y tipos de puertos (Físicos y Lógicos)
 - 5.4.2. Puertos de Entrada, Salida y Entrada/Salida Práctica de Laboratorio #12: “Manejo de Archivos en MatLab” Práctica de Laboratorio #13: “Práctica de adquisición y manejo de señales a través de un puerto” 1.3.2.

NOTA. - Todas LAS PRÁCTICAS son de 2 hrs/sem/mes.



UNIVERSIDAD AUTÓNOMA DE SINALOA
FACULTAD DE INGENIERÍA MOCHIS
LICENCIATURA EN INGENIERÍA EN PROCESOS INDUSTRIALES



PROGRAMA DE ESTUDIO

5. ACTIVIDADES PARA DESARROLLAR LAS COMPETENCIAS

Actividades del docente:

- Diseñar un plan de estudios que incluya objetivos de aprendizaje, temas a cubrir, recursos necesarios y actividades a realizar.
- Explicar conceptos teóricos de programación y algoritmos, utilizando métodos de enseñanza variados (clases magistrales, demostraciones, etc.) para captar la atención de los estudiantes.
- Diseñar y aplicar evaluaciones (exámenes, pruebas, proyectos) para medir el progreso y comprensión de los estudiantes, brindando retroalimentación oportuna.
- Utilizar herramientas tecnológicas y plataformas de programación que mejoren el proceso de enseñanza-aprendizaje, así como mantenerse actualizado sobre nuevas tecnologías y lenguajes de programación.
- Abordar la importancia de la ética en la programación, discutiendo casos prácticos y promoviendo un comportamiento responsable en el uso de tecnologías.
- Reflexionar sobre su propia práctica docente, buscando oportunidades de mejora en su enseñanza y en la forma de interactuar con los estudiantes.

Actividades del estudiante:

- ❖ Participar activamente en las sesiones teóricas y prácticas, tomando notas y haciendo preguntas para aclarar dudas.
- ❖ Leer y estudiar los recursos proporcionados por el docente, incluyendo libros, artículos y documentación relacionada con la programación y algoritmos.
- ❖ Trabajar en proyectos individuales o en equipo que integren los conocimientos adquiridos, permitiendo la aplicación práctica de la programación en contextos industriales.
- ❖ Participar en actividades grupales y proyectos colaborativos, compartiendo ideas y aprendiendo de las experiencias de sus compañeros.
- ❖ Evaluar su propio progreso y reflexionar sobre las habilidades adquiridas, identificando áreas de mejora.
- ❖ Mantenerse actualizado sobre las tendencias y avances en programación, explorando nuevos lenguajes y herramientas que puedan ser útiles en su formación.

6. EVALUACIÓN DE LAS COMPETENCIAS

6.1. Criterios de desempeño	6.2 Portafolio de evidencias
- Comprensión Teórica - Calidad del Código - Resolución de Problemas - Proyectos y Tareas - Participación y Colaboración - Pruebas y Exámenes	- Documentación de ejercicios realizados, incluyendo el enunciado del problema, el código escrito y la explicación de la solución. - Detalles sobre los proyectos realizados, incluyendo el objetivo, el enfoque utilizado, el código fuente, documentación y una reflexión sobre el proceso y los resultados. - Incluir cualquier retroalimentación recibida de los docentes sobre trabajos, proyectos o desempeño en clase.



UNIVERSIDAD AUTÓNOMA DE SINALOA
FACULTAD DE INGENIERÍA MOCHIS
LICENCIATURA EN INGENIERÍA EN PROCESOS INDUSTRIALES



PROGRAMA DE ESTUDIO

	- Registro de actividades de colaboración en grupo, incluyendo proyectos conjuntos, discusiones en clase y cualquier evaluación relacionada con la participación.			
6.3. Calificación y acreditación:				
Parcial: Exámenes Participación en Clase Ejercicios y Tareas Proyectos Parciales Dinámicas de Grupo Informe de Autoevaluación		Final: Evaluación Teórica (30%) Evaluación Práctica (40%) Evaluación Colaborativa (20%) Autoevaluación y Reflexión (10%)		
7. FUENTES DE INFORMACIÓN				
Bibliografía básica				
Autor(es)	Título	Editorial	Año	URL o biblioteca digital donde está disponible
Norton, Peter	Introducción computación	Mc Graw Hill	2006	
Joyanes Aguilar, Luis	Fundamentos de Programación	Mc Graw Hill	2008	
J. Glenn Brookshear	Introducción computación	Pearson	2012	
8. PERFIL DEL DOCENTE				
Licenciatura o grado en Ingeniería en Sistemas, Ingeniería en Computación, Ciencias de la Computación o campos afines. Conocimiento profundo de varios lenguajes de programación relevantes (por ejemplo, Python, Java, C++, etc.). Comprensión sólida de algoritmos y estructuras de datos, así como su aplicación en la resolución de problemas. Habilidad para explicar conceptos complejos de manera clara y accesible, tanto en forma oral como escrita. Preferiblemente, contar con experiencia profesional en el campo de la programación o el desarrollo de software, lo que permite integrar ejemplos prácticos en la enseñanza.				