



UNIVERSIDAD AUTÓNOMA DE SINALOA
FACULTAD DE CIENCIAS DE LA TIERRA Y EL ESPACIO
LICENCIATURA EN ASTRONOMÍA
PROGRAMA DE ESTUDIOS

1. DATOS DE IDENTIFICACIÓN			
UNIDAD DE APRENDIZAJE O MÓDULO	Programación		
Clave:	(Pendiente)		
Ubicación	II SEMESTRE - ASTRONOMÍA		
Horas y créditos:	Teóricas: 40	Prácticas: 40	Estudio Independiente: 16
	Total de horas: 96	Créditos: 6	
Competencia (s) del perfil de egreso a las que aporta:	El alumno aprende a solucionar problemas matemáticos mediante lenguajes de programación para automatizar procesos del campo profesional, con base en habilidades de razonamiento lógico-computacional.		
Componentes de la competencia que se desarrollan en la Unidad de Aprendizaje:	El alumno, comprende y aplica los fundamentos básicos de Programación, así como la sintaxis de programación relacionados con software especializado en Astronomía, además maneja la resolución de problemas matemáticos computacionales, y estructura módulos para la automatización de procesos estadísticos.		
Unidades de aprendizaje relacionadas:	Tecnologías de la Información y Comunicación, Programación II		
Responsables de elaborar el programa:	M.C. Thania Roxaana Félix González, M.C. José De Jesús Uriarte Adrián, M.C. Miguel Armando López Beltrán, L.I. Abel Cota Dimas.	Fecha: Enero de 2018	
Responsables de actualizar el programa:	M.C. Thania Roxaana Félix González, M.C. José De Jesús Uriarte Adrián, M.C. Miguel Armando López Beltrán, L.I. Abel Cota Dimas.	Fecha: Enero de 2023	
2. PROPÓSITO			
Al finalizar el curso, el alumno será capaz de analizar, diseñar, probar y construir programas aplicando los nuevos paradigmas de ingeniería de software.			
3. SABERES			
Teóricos:	<ul style="list-style-type: none">• Obtener los conocimientos básicos sobre algoritmos, así como, los tipos de datos existentes y expresiones algorítmicas.• Comprender las diferentes estructuras algorítmicas.• Obtener las herramientas básicas de la metodología de programación en un lenguaje de programación de alto nivel.• Comprender el concepto de programación modular empleado en un lenguaje de programación de alto nivel.• Entender los tipos de datos estructurados.		
Prácticos:	<ul style="list-style-type: none">• Aplicar los conocimientos básicos sobre algoritmos, así como, sobre los tipos de datos existentes y su aplicación en las diferentes sentencias y expresiones algorítmicas.• Construir programas en pseudocódigo aplicando las estructuras algorítmicas.• Aplicar las herramientas básicas de la metodología de programación en un lenguaje de programación de alto nivel.• Crear programas en pseudocódigo aplicando el concepto de programación modular para luego codificarlos en un lenguaje de programación de alto nivel.• Desarrollar programas aplicando los tipos de datos estructurados.		
Actitudinales:	<ul style="list-style-type: none">• Valorar el papel de la ciencia y la tecnología en la programación dirigida a objetos.• Disposición al trabajo colectivo.• Cultivar la disciplina de la lectura científica.• Desarrollar la ética profesional.		

	<ul style="list-style-type: none"> • Reflexividad ante las diferentes propuestas técnicas y metodológicas. • Atención a la actualización profesional.
4. CONTENIDOS	
1. INTRODUCCIÓN A LA PROGRAMACIÓN <ol style="list-style-type: none"> 1.1. Conceptos básicos 1.2. Concepto de lenguaje 1.3. Clasificación de los lenguajes 1.4. Compiladores e intérpretes 1.5. Paradigmas de la Programación 1.6. Principales herramientas para el análisis y diseño. 2. FUNDAMENTOS DE PROGRAMACIÓN <ol style="list-style-type: none"> 2.1. Solución de problemas aplicando la computadora. 2.2. Algoritmos. 2.3. Tipos de datos. 2.4. Constantes y variables. 2.5. Sentencias y expresiones. 2.6. Estructuras Secuenciales. 2.7. Estructuras Selectivas. 2.8. Estructuras Repetitivas. 3. CODIFICACIÓN DE PSEUDOCODIGOS <ol style="list-style-type: none"> 3.1. Tipos de datos. 3.2. Constantes y Variables. 3.3. Conversión entre tipos de datos. 3.4. Expresiones y Operadores. 3.5. Operaciones de entrada/salida. 3.6. Sentencias de control. 3.7. Estructuras secuenciales. 3.8. Estructuras selectivas. 3.9. Estructuras repetitivas. 4. PROGRAMACIÓN ESTRUCTURADA. <ol style="list-style-type: none"> 4.1. Técnicas de programación estructurada. 4.2. Funciones. 4.3. Procedimientos. 4.4. Codificación en lenguaje de programación 	
ACTIVIDADES PARA DESARROLLAR LAS COMPETENCIAS	
Actividades sugeridas para el docente: Exponer del tema de manera clara, realizar lecturas de investigación para complementar la comprensión del tema, transferir contenidos temáticos mediante los medios electrónicos, así como plantear y solucionar problemas concretos.	
Actividades sugeridas para el estudiante: Analizar y debatir de manera grupal sobre el tema, atender trabajos de investigación y redacción de resúmenes, proponer la solución a problemas extra clase y trabajar de manera colectiva en exposiciones.	
6. EVALUACIÓN DE LAS COMPETENCIAS	
6.1. Evidencias	Indicadores de calidad generales
Resolución de ejercicios en clase en forma individual y grupal, resolución de ejercicios de tarea, debate en torno a las dudas de los estudiantes, resolución de problemas utilizando herramientas de cómputo y examen.	La comprensión y profundización de conceptos teóricos, las diferentes capacidades de trabajo colectivo intelectual y práctico, de exposición y dominio temático, de análisis, redacción y síntesis de la investigación bibliográfica y de planeación y desarrollo metodológico en la solución de problemas.

6.3. Calificación y acreditación:

Asistencia y participación 20%, Tareas 30%, Examen 50%

7. FUENTES DE INFORMACIÓN**Básica:**

AYALA SAN MARTIN, GERARDO. (1987) Computación I, Introducción a la Computación. Edit. Porrúa, Universidad Nacional Autónoma de México. México, D.F. Primera edición.

LEVINE GUTIERREZ, GUILLERMO. (1994) Introducción a la Computación y a la Programación Estructurada. Edit. McGraw-Hill. Primera edición.

JOYANES AGUILAR, LUIS. (1996) Fundamentos de Programación, Algoritmos y Estructura de datos. Edit. McGraw-Hill. Madrid, España. Segunda edición.

JOYANES AGUILAR, LUIS. (1997) Metodología de la Programación, Diagramas de flujo algoritmos y programación estructurada. Edit. McGraw-Hill. México, D.F. Primera edición.

LÓPEZ ROMÁN, LEOBARDO (2006) Metodología de la Programación Orientada a Objetos. Edit. Alfaomega, México D.F. Primera Edición.

Complementaria:

LÓPEZ ROMÁN, LEOBARDO. (2003) Programación Estructurada, un enfoque algorítmico. Edit. Alfaomega, México D.F. Segunda Edición.

SCHILDT HERBERT (2003), C#. Manual de referencia. Mc Graw Hill, Primera Edición.

CAIRO BATTISTUTTI, OSVALDO. (2006) Metodología de la Programación, Algoritmos, Diagramas de Flujo y Programas. Edit. Alfaomega, México D.F Tercera edición.

CEBALLOS, Fco. Javier (2008), C# Lenguajes y Aplicaciones, Edit. Alfaomega, México D.F, Segunda Edición.

8. PERFIL DEL PROFESOR:

El profesor debe de contar con posgrado en el área de las Ciencias de la Información con orientación en Programación o en alguna disciplina de la informática. Debe de contar con experiencia docente y en trabajos de investigación o aplicación de los métodos de programación para la generación de información geográfica.