



**UNIVERSIDAD AUTÓNOMA DE SINALOA
FACULTAD DE CIENCIAS FÍSICO MATEMÁTICAS
CARRERA: LICENCIATURA EN MATEMÁTICAS**

PROGRAMA DE ESTUDIOS

1. DATOS DE IDENTIFICACIÓN			
UNIDAD DE APRENDIZAJE	COMPUTACIÓN		
Clave:	404		
Semestre:	I semestre		
Eje Curricular:	<input checked="" type="checkbox"/> Básico <input type="checkbox"/> Profesionalizante <input type="checkbox"/> Acentuación		
Área:	<input checked="" type="checkbox"/> Física-Matemática <input type="checkbox"/> Cs. Sociales y Humanidades <input type="checkbox"/> Idiomas <input type="checkbox"/> Básico Profesional <input type="checkbox"/> Profesional		
Horas y créditos:	Teóricas: 64	Prácticas:	Estudio Independiente:
	Horas por semana: 4	Créditos: 8	
	Total por semestre: 64		
Tipo de curso:	Teórico (<input type="checkbox"/>)	Teórico-práctico (<input checked="" type="checkbox"/>)	Práctico
Competencias del perfil de egreso a la que aporta			
Unidades de aprendizaje relacionadas	Cálculo I, Cálculo II, Estadística, Física General, Introducción al Cálculo, Matemáticas Discretas, Probabilidad.		
Responsables de elaborar y/o actualizar el programa:	M.C. Alonso Núñez Páez Dr. José Ángel Islas Anguiano		
Fecha de:	Elaboración: Febrero 2012	Actualización: Marzo 2019	
2. PROPÓSITO			
Programar y ejecutar algoritmos computacionales en el lenguaje C para resolver problemas matemáticos. Manejar herramientas computacionales para agilizar procesos matemáticos.			
3. SABERES			
Teóricos:	<ul style="list-style-type: none"> - Conoce la historia de la computación - Conoce los fundamentos de la programación - Conoce el lenguaje de programación 		

Prácticos:	<ul style="list-style-type: none"> - Desarrolla e implementa algoritmos - Implementa algoritmos en C
Actitudinales:	<ul style="list-style-type: none"> - Actitud de participación en la solución de ejercicios - Cultiva el autoaprendizaje - Actitud reflexiva en la apropiación de nuevos conceptos - Aprecia la potencialidad de la programación como puente para el desarrollo de las ciencias
4. CONTENIDO TEMÁTICO	
<p>1. Introducción (6h)</p> <p>1.1. Reseña histórica de la computación. 1.2. La computadora como herramienta y objeto de estudio. 1.3. Lenguajes de programación. 1.3.1. Arquitectura Von Newman. 1.3.1.1. Lenguajes de programación.</p> <p>2. Algoritmos: descripción y notación algorítmica (14h)</p> <p>2.1. Aproximación intuitiva a los algoritmos mediante ejemplos. 2.2. Definición de un algoritmo. 2.3. Formas de describir un algoritmo: Pseudocódigo y PSe int. 2.4. Desarrollo e implementación de un algoritmo.</p> <p>3. El lenguaje C (22h)</p> <p>3.1. Conceptos básicos de C 3.2. Operadores y expresiones 3.2.1. Operadores aritméticos, unarios, relacionales y lógicos. 3.2.2. Operadores de asignación. 3.2.3. Operador condicional. 3.2.4. Funciones de biblioteca. 3.3. Entrada y salida de datos 3.3.1. Introducción. 3.3.2. Funciones: getchar, putchar, scanf, printf gets y puts. 3.4. Instrucciones de control 3.4.1. Instrucciones if –else. 3.4.2. Instrucciones while, do while, for. 3.4.3. Instrucciones switch, break, continue, goto.</p> <p>4. Funciones (6h)</p> <p>4.1. Definición de una función 4.2. Acceso a una función. 4.3. Prototipos de funciones. 4.4. Paso de argumentos a una función. 4.5. Recursividad.</p> <p>5. Estructura de un programa (4h)</p> <p>5.1. Tipos de almacenamiento. 5.2. Variables automáticas, externas (globales) y estáticas. 5.3. Programas de varios archivos.</p>	

6. Arreglos (4h)

- 6.1 Definición de un arreglo.
- 6.2 Arreglos bidimensionales.

7. Apuntadores (4h)

- 7.1 Conceptos básicos
- 7.2 Asignación dinámica de memoria

8. Estructuras (4h)

- 8.1 Definición de una estructura
- 8.2 Componentes de una estructura
- 8.3 Tipos de datos
- 8.4 Procesamiento de una estructura.

5. ACCIONES ESTRATÉGICAS PARA EL APRENDIZAJE

Sensibilización y atención:

- Realizar una exposición introductoria de los temas en cada unidad, haciendo mención del contexto histórico en que los conceptos fueron desarrollados, así como de los problemas teóricos o tecnológicos que se pueden resolver con los temas que se verán en dicha unidad temática.
- Explicar las técnicas para resolver los problemas teóricos y/o prácticos que contribuyan a comprender la temática de la unidad.
- Recomendar lectura previa de temas selectos, para crear discusiones y debates en torno al tema.

Estrategias y técnicas de aprendizaje:

- Aprendizaje basado en problemas.
- Aprendizaje colaborativo en la resolución de ejercicios y en exposiciones.
- Método de proyectos.
- Exposición guiada.

6. EVALUACIÓN DEL APRENDIZAJE

6.1. Evidencias de aprendizaje	6.2. Criterios de desempeño	6.3. Calificación y acreditación
	<ul style="list-style-type: none">- Exámenes por unidad: Descripción correcta de los conceptos importantes de los temas y procedimientos y solución correcta de problemas-Examen proyecto final: Desarrollar un proyecto de interés matemáticos o algún área a fin. Deberá ser capaz de modificar el proyecto final a petición del instructor en el tiempo dedicado a la evaluación.	<ul style="list-style-type: none">20% Tareas30%Examen Cap 1 al 3.50%Examen proyecto final.

Fuentes de Información Básica:

Algorithms

Sedgewick, R.
Addison Wesley

Programación en C 2ª ed.

Byron S. Gottfield
McGraw – Hill

Estructura de Datos y Algoritmos

Weiss, M. A.
Addison Wesley

Data Structures and Algorithms

Aho, A. V. , Hopcroft J. E. and Ullman J. D.
Addison Wesley

Programación en Microsoft C para IBM, PCs y Compatibles: Introducción y Técnicas Avanzadas de Programación.

Lafore, Robert
Anaya Multimedia. Madrid

C/C++: Curso de Programación 2ª ed.

Ceballos Sierra, Francisco J.
Alfaomega

Fuentes de Información Complementaria:

Project Euler

<https://projecteuler.net/>

8. PERFIL DEL PROFESOR:

Licenciado en Matemáticas con conocimientos sólidos en computación o Licenciado en Computación.

Conoce y aplica adecuadamente la programación para resolver problemas de interés matemático.

Posee formación sólida en computación y matemáticas, que le permite conectar los saberes del curso con otras asignaturas.

Tiene habilidades didácticas de enseñanza y evaluación del aprendizaje.