



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



PROGRAMA DE ESTUDIOS

1. DATOS DE IDENTIFICACIÓN			
UNIDAD DE APRENDIZAJE	COMPUTACIÓN I		
Clave:	3418		
Semestre:	IV		
Eje Curricular:	<input checked="" type="checkbox"/> Básica <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: 30	Prácticas: 20	Estudio Independiente: 10
	Total de horas: 60		Créditos: 8
Tipo de curso:	<input type="checkbox"/> Teórico	<input checked="" type="checkbox"/> Teórico-práctico	<input type="checkbox"/> Práctico
Competencias del perfil de egreso a la que aporta	<p>Analiza e identifica variables y parámetros, en la solución secuencial de problemas, desde la metodología de algoritmos computacionales.</p> <p>Construye algoritmos computacionales, para su programación en distintos lenguajes, desde la metodología de pseudocódigo apropiado.</p> <p>Elabora programas en el lenguaje Fortran, para algoritmos específicos, desde la metodología de lenguaje de programación.</p>		
Unidades de aprendizaje relacionadas	Álgebra superior, Geometría euclidiana, Álgebra lineal I.		
Responsables de elaborar y/o actualizar el programa:	MC. Alonso Núñez Páez		
Fecha de	Elaboración: Feb-2012		Actualización: Enero-2018
2. PROPÓSITO			
Construcción de algoritmos computacionales, codificación de algoritmos en el lenguaje de programación Fortran, su compilación y ejecución.			
3. SABERES			
Teóricos:	<ul style="list-style-type: none"> - Identifica las variables y parámetros y sus interrelaciones en la construcción de algoritmos, para la resolución de problemas, en pseudocódigo apropiado. - Análisis, elaboración e interpretación de algoritmos. - Conoce el pseudocódigo adecuado para describir los algoritmos. - Conoce el lenguaje de programación Fortran, así como las etapas de compilación y ejecución. 		

Prácticos:	<ul style="list-style-type: none"> - Conoce y usa el software para la prueba y corrección de algoritmos en seudocódigo apropiado. - Conoce y usa el software apropiado para la codificación, compilación y ejecución, adecuados para el lenguaje de programación Fortran.
Actitudinales:	<ul style="list-style-type: none"> - Actitud de participación en la solución de ejercicios. - Cultiva el auto aprendizaje. - Actitud reflexiva en la apropiación de nuevos conceptos. - Valora la potencialidad de la programación.

4. CONTENIDO TEMÁTICO

1. **Introducción.**

- 1.1. Reseña histórica de la computación.
- 1.2. La computadora como herramienta y objeto de estudio.

2. **Algoritmos: descripción y notación algorítmica.**

- 2.1. Aproximación intuitiva a los algoritmos mediante ejemplos.
- 2.2. Definición de un algoritmo.
 - 2.2.1. Máquinas para aplicar algoritmos.
 - 2.2.2. Elementos para aplicar algoritmos.
- 2.3. Formas de describir un algoritmo.
 - 2.3.1. Lenguaje natural.
 - 2.3.2. Organigramas (lenguajes de flujo)
 - 2.3.3. Seudocódigo.
 - 2.3.4. Lenguajes de programación.
- 2.4. Desarrollo e implementación de un algoritmo.
 - 2.4.1. Conceptos fundamentales: acciones, variables, información, estados, Datos/resultados, léxico, tipos de datos y operadores.
 - 2.4.2. Notación algorítmica: variables, tipos de datos, literales, expresiones, Asignación, entrada/salida.

3. **Lenguajes de programación.**

- 3.1. Arquitectura Von Newman.
 - 3.1.1. Componentes y relación entre ellos.
 - 3.1.2. Código binario.
 - 3.1.3. Representación de datos en código binario: caracteres, valores lógicos, números enteros, números reales.
- 3.2. Lenguajes de programación.
 - 3.2.1. Lenguaje máquina.
 - 3.2.2. Lenguaje ensamblador.
 - 3.2.3. Lenguaje de alto nivel.

4. **Introducción al lenguaje FORTRAN (Fortran).**

- 4.1. Historia del lenguaje Fortran.
- 4.2. Introducción a la notación Fortran (Sintaxis).
 - 4.2.1. Variables.
 - 4.2.2. Tipos de datos.
 - 4.2.3. Literales.
 - 4.2.4. Expresiones.
 - 4.2.5. Asignación.

4.2.6. Entrada/Salida.

5. Estructura de un programa en Fortran.

- 5.1. Declaración de variables.
 - 5.1.1. Declaración implícita y explícita.
 - 5.1.2. Declaración de cadenas de caracteres.
 - 5.1.3. Declaración de vectores y matrices.
- 5.2. Entrada/Salida.
 - 5.2.1. Sentencias de Entrada/Salida.
 - 5.2.2. Códigos de formatos.
- 5.3. Estructuras de control.
 - 5.3.1. Estructura secuencial.
 - 5.3.2. Estructura alternativa.
 - 5.3.2.1. Simple.
 - 5.3.2.2. Compuesta.
 - 5.3.2.3. Múltiple.
 - 5.3.3. Estructura repetitiva.
 - 5.3.3.1. Desde – Hasta.
 - 5.3.3.2. Mientras.
 - 5.3.3.3. Hasta.

6. Funciones y subrutinas.

- 6.1. Funciones.
 - 6.1.1. Funciones intrínsecas.
 - 6.1.2. Funciones definidas por el usuario.
- 6.2. Subrutinas o procedimientos.

7. Recursividad.

- 7.1. Diseño de algoritmos recursivos.
- 7.2. Sintaxis de subprogramas recursivos.
- 7.3. Algoritmos recursivos contra iterativos.

8. Archivos

- 8.1. Apertura y cierre de archivos.
- 8.2. Lectura/Escritura de archivos.

5. ACCIONES ESTRATÉGICAS PARA EL APRENDIZAJE

Motivación al tema:

- Recomendar lectura previa de temas selectos de cada unidad, para crear discusiones y debates en torno al tema.
- Realizar una exposición introductoria de los temas en cada unidad, estableciendo los conceptos fundamentales y sus propiedades.
- Explicar las técnicas para resolver los problemas teóricos y/o prácticos que contribuyan a comprender la temática de la unidad.

En la plataforma virtual o redes sociales:

- Transferencia de información al alumno de algunos temas concretos.
- Entrega de tareas.
- Apertura de foros de discusión y seguimiento a ellos.

Estrategias y técnicas de aprendizaje:

- Aprendizaje basado en problemas.
- Aprendizaje colaborativo en la resolución de ejercicios.

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 - Prácticas de ejercicios 	<ul style="list-style-type: none"> - Exámenes por unidad: Descripción completa de los conceptos importantes de los temas y procedimientos, así como solución correcta de problemas. - Prácticas de ejercicios: 60% planteamiento del problema, 20% Procedimiento y 20 % Resultados. 	80% Cuatro exámenes. 20% Prácticas de ejercicios.

7. FUENTES DE INFORMACIÓN

Fuentes de Información Básica:

Fortran 90/95 Explained
Oxford Univ. Press

Introducing FORTRAN 95
Chivers, I. and Sleightholme, J.
Spring-Verlang London Limited

Lenguaje de programación Fortran 90: incluye Fortran 95.
García Merayo F.
Paraninfo

Fortran 90 Handbook.
Adams, Brainerd, Martin, Smith, Wagener.
McGraw-Hill

Fortran 90/95 for scientists and engineers.
Chapman S. J.
McGraw-Hill.

Algorithms
Sedgewick, R.
Addison Wesley

Estructura de Datos y Algoritmos
Weiss, M. A.
Addison Wesley

Fuentes de Información Complementaria:

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

Algoritmos y Estructuras de Datos (2 Vols.)

García, M. y Cervera López.

Diego Marín Librero-Editor

Fortran 90 Course Notes. 1997.

Marshall, A. C. and Morgan, J. S. and Schonfelder J. L.

<http://www.liv.ac.uk/HPC/F90page.html>

Fortran 90 Tutorial.

Dodson, Z. A

<http://www.glue.umd.edu/~nsw/fortran/matthias/tutorial.html> 1994

8. PERFIL DEL PROFESOR:

- Posee formación sólida en programación, de manera que le permita conectar los saberes del curso con otras asignaturas.
- Conoce y aplica adecuadamente la teoría base para la elaboración y programación de algoritmos.
- Integra eficientemente las Tecnologías de la Información y Comunicación (TIC) en clase.
- Utiliza software específicos para la resolución de problemas sobre las temáticas del curso.