



1. DATOS DE IDENTIFICACIÓN					
UNIDAD DE APRENDIZAJE O MÓDULO:	LABORATORIO DE ALGORITMOS Y LÓGICA COMPUTACIONAL				
Clave:					
Ubicación:	SEMESTRE I	Área: Profesionalizante			
Horas y créditos:	Teóricas: 0	Prácticas: 50	Estudio Independiente: 46		
	Total de horas: 96		Créditos: 6		
Competencia(s) del perfil de egreso al que aporta:	<p>Genéricas:</p> <p>CG1. Desarrolla su potencial intelectual para generar el conocimiento necesario en la resolución de problemas y retos, tanto de su vida individual y como parte de una comunidad, con sentido de pertinencia, identidad y empatía.</p> <p>CG10. Asume con responsabilidad y ética el manejo de las Tecnologías del Aprendizaje y el Conocimiento y es capaz de re conducir las Tecnologías de la Información y Comunicación para la adquisición y actualización del conocimiento de manera permanente para su vida y su profesión.</p> <p>Específicas:</p> <p>CE13. Aplica conocimientos teóricos y prácticos, en conjunto con las metodologías para la buena construcción de programas y sistemas de software, considerando su análisis y diseño, planeación, procesos y controles dentro de escenarios de pruebas para asegurar confiabilidad, funcionalidad, costo, seguridad, facilidades de mantenimiento y otros aspectos relacionados.</p>				
Unidades de aprendizaje relacionadas:	Matemáticas discretas, programación I, programación II, sistemas digitales, inteligencia artificial, sistemas operativos, programación orientada a objetos, ingeniería de software II, estructura de datos.				
Responsable(s) de elaborar el programa:	L.I. MANUEL DE JESÚS RODRÍGUEZ GUERRERO		Fecha: 5 de Julio del 2011		
Responsable(s) de actualizar el programa:			Fecha:		
2. PROPÓSITO					
Conocer y analizar los elementos necesarios para poder desarrollar soluciones a problemas que se pueden resolver con algoritmos computacionales y que le permitan obtener las habilidades básicas para introducirse en la ingeniería de software.					



PROGRAMA DE ESTUDIO

3. SABERES	
Teóricos:	Explicará qué es un algoritmo mediante el análisis de sus características e identificará que es un lenguaje de programación para obtener los conocimientos previos necesarios para el estudio de la programación. Usará la estructura secuencial para la solución de problemas. Aplicará el concepto de repeticiones fijas y variables, así como anidadas a la solución de problemas en un ambiente gráfico. Usará procedimientos para mejorar la estructura de sus programas. Usará las nuevas instrucciones para construir procedimientos recursivos y aplicará la recursividad en la solución de problemas. Utilizará procedimientos para comprender el concepto de parámetros, variable y constante y aplicarlos en la solución de problemas.
Prácticos:	Resolver la evaluación diagnóstica. realizar un trabajo de investigación sobre los algoritmos. Investigar a través de diferentes medios las definiciones de programa y lenguaje de programación. Participar en la resolución de un algoritmo propuesto por el docente, externando sus ideas ante el grupo. Analizar el problema para darle la solución correcta de acuerdo a las características de los algoritmos. Identificar las reglas para la escritura de repeticiones fijas. Revisar los ejemplos propuestos por el docente acerca de repeticiones fijas simples (no anidadas). Identificar las reglas para la escritura de repeticiones condicionadas. revisar los ejemplos propuestos por el docente acerca de repeticiones condicionadas (no anidadas). Resolver los ejercicios de aplicación de las reglas para escribir variables y constantes en un programa. Identificar los elementos que componen a una nueva instrucción, su ubicación dentro del código, como escribirlas y como usarlas. analizar los ejemplos del profesor del uso de nuevas instrucciones. Resolver ejercicios de elaboración y uso de nuevas instrucciones. Identificar los elementos que componen a un procedimiento recursivo. analizar las semejanzas y diferencias de una misma nueva instrucción escrita con un mientras y escrita de manera recursiva. Observar el funcionamiento de los procedimientos demostrados por el profesor y hacer las pruebas de escritorios necesarias. Escribirá y probará en la computadora los ejemplos planteados en el salón y por equipos solucionarán.
Actitudinales:	Trabajo de programación en equipo mediante una metodología establecida. autocrítica de redacción de documentos. Autocrítica de presentaciones orales. crítica constructiva al trabajo de otros. Aprendizaje de las exposiciones de otros.
4. CONTENIDOS	
<p>Unidad 1 Introducción</p> <ul style="list-style-type: none">1.1 Introducción1.2 Definición de algoritmo1.3 Características de un algoritmo1.4 Tipos de algoritmos1.5 Algoritmos cotidianos1.6 Lenguajes algorítmicos <p>Unidad 2 Metodología para la solución de problemas por medio de la computadora</p> <ul style="list-style-type: none">2.1 Definición del problema2.2 Análisis del problema	



PROGRAMA DE ESTUDIO

2.3 Diseño de la solución

2.4 Codificación

2.5 Pruebas y depuración

2.6 Documentación

2.6.1 Mantenimiento

Unidad 3 Entidades primitivas para el desarrollo de algoritmos

3.1 Tipos de datos

3.1.1 Tipos de datos simples

3.1.2 Tipos de datos estructurados

3.2 Identificadores

3.3 Constantes

3.4 Variables y su clasificación

3.4.1 Expresiones

3.5 Operadores y su jerarquía

3.5.1 Operadores aritméticos

3.5.2 Operadores relacionales

3.6 Operadores lógicos

Unidad 4 Elaboración de algoritmos en DFD

4.1 Conocimiento del entorno

4.2 Primer diagrama en DFD

Unidad 5 Estructuras secuenciales de control de algoritmos

5.1 Asignación

5.2 Entrada

5.3 Salida

5.4 Prácticas de estructuras secuenciales

Unidad 6 Estructuras condicionales de control de algoritmos

6.1 Estructuras condicionales simples

6.2 Estructuras condicionales múltiples

6.3 Prácticas de estructuras condicionales

Unidad 7 Funciones predefinidas

7.1 Prácticas de funciones en DFD

Unidad 8 Estructuras cíclicas de control de algoritmos

8.1 Ciclo Para

8.2 Prácticas de ciclo Para

8.3 Ciclo Mientras que

8.4 Prácticas de ciclo mientras

Unidad 9 Arreglos

9.1 Vectores

9.2 Prácticas de vectores

9.3 Matrices

9.4 Prácticas de Matrices



PROGRAMA DE ESTUDIO

Unidad 10 Elaboración de algoritmos en PSeInt

- 10.1 Conocimiento del entorno
- 10.2 Estructuras condicionales múltiples en PSeInt
- 10.3 Ciclo Repetir Hasta en PSeInt
- 10.4 Prácticas de algoritmos en PSeInt

5. ACTIVIDADES PARA DESARROLLAR LAS COMPETENCIAS

Actividades del docente:

- Establecer el encuadre del curso al inicio de este.
- Asistir puntualmente a clases y cumplir el tiempo indicado.
- Preparar material didáctico para las clases.
- Aclarar las dudas de los alumnos.
- Asesorar a los alumnos cuando lo soliciten.
- Evaluar el aprendizaje conforme a lo acordado en el encuadre.
- Entregar resultados de las evaluaciones en tiempo y forma.

Actividades del estudiante:

- ❖ Asistir y llegar puntualmente a clases.
- ❖ Tener como mínimo, el 80 % de asistencias. participar de manera activa en los procesos de enseñanza y aprendizaje.
- ❖ Prepararse para realizar las evaluaciones que se establezcan.

6. EVALUACIÓN DE LAS COMPETENCIAS

6.1. Criterios de desempeño	6.2 Portafolio de evidencias
<ul style="list-style-type: none">• Lecturas obligadas y estudio individual.• Resolución de prácticas, problemas o ejercicios propuestos por profesor.• Preparación de trabajos individuales.• Exposiciones de trabajos en clase.• Tutorías grupales individuales y Virtuales.• Búsquedas de información en Bibliotecas, hemerotecas, internet...	<ul style="list-style-type: none">• Evaluación teórica.• Elaboración de ejercicios prácticos en clase.• Trabajos de investigación (artículos, libros, internet, etc.).• Glosario de conceptos de la materia.• Mapa mental de los contenidos de la materia.

6.3. Calificación y acreditación:

Parcial:	Final:
<ul style="list-style-type: none">• Examen escrito 50%• Participación en clase 10%• Asistencia 10%	<ul style="list-style-type: none">• Promedio de las evaluaciones parciales.



PROGRAMA DE ESTUDIO

- | | |
|--------------------------------|--|
| • Portafolio de evidencias 30% | |
|--------------------------------|--|

7. RECURSOS DIDÁCTICOS

8. FUENTES DE INFORMACIÓN

Bibliografía básica

Autor(es)	Título	Editorial	Año	URL o biblioteca digital donde está disponible
Guillermo Correa Uribe	Desarrollo de Algoritmos Y Sus Aplicaciones en Basic, Pascal, Cobol	McGraw-Hill.	1998	
Francisco Javier Ceballos	Curso de Programación con C	Macrobit.	1990	

Bibliografía complementaria

Autor(es)	Título	Editorial	Año	URL o biblioteca digital donde está disponible
Manuel Abellanas y Dolores Lodares	Análisis de algoritmos y teoría de grafos	Ra-Ma	1990	
Selim G. Akl	The Desing and Analysis of Parallel Algorithms	Prentice-Hall	1989	

9. PERFIL DEL DOCENTE

El perfil del profesor debe ser un Lic. o ingeniero en sistemas, informática, software, sistemas computacionales, electrónica o afín.