



**1. Datos Generales de la asignatura**

<b>Nombre de la asignatura:</b>	Lógica de Programación
<b>Clave de la asignatura:</b>	NAF-0912
<b>SATCA<sup>1</sup>:</b>	3-2-5
<b>Carrera:</b>	Ingeniería en Nanotecnología

**2. Presentación**

**Caracterización de la asignatura.**

Actualmente el uso de computadoras ha tenido gran impacto en áreas diseño; entre otras cosas, permite modelar sistemas, y procesar información a velocidades elevadas.

Sin importar el área de especialidad, se deben tener conocimientos básicos de programación para utilizar paquetes especializados o para desarrollar alguna aplicación para resolver problemas reales. Estos conocimientos también permiten comprender como la computadora lleva a cabo sus operaciones, y así poder proponer diferentes soluciones a dichos problemas.

Entre algunas aplicaciones para los Ingenieros en Nanotecnología se encuentra la caracterización y modelado molecular en Ciencia de los Materiales, modelado de estructuras de transición, etc.

La materia aporta conocimientos y habilidades para las materias de Análisis Numérico e Introducción al Modelado por Computadora.

**Intención didáctica.**

Los contenidos deben ser abordados de tal manera que el estudiante vaya sumando los conocimientos y habilidades en el desarrollo de competencias, que le permitan usar la computadora y la programación como herramienta para la resolución de problemas en su área de especialidad.

Se organiza el temario de manera que el estudiante haga una transición ágil y dinámica de los conceptos básicos de computación, a la programación.

El alcance que debe tener es que el estudiante comprenda como los paquetes de cómputo procesan operaciones, pueda hacer planteamiento de problemas para su solución con herramientas y técnicas de programación, y que pueda identificar variables y procesos para aplicarlos en software especializado de procesamiento y simulación para su área de especialidad.

Dentro de la estructura planteada en el contenido temático, el primer tema tiene la intención de contextualizar al estudiante, introducirlo a los conceptos básicos necesarios para que pueda

<sup>1</sup> Sistema de asignación y transferencia de créditos académicos

desarrollar las competencias propuestas en el resto del plan de estudios.

El poder desarrollar sistemáticamente una propuesta solución para un problema, es la finalidad del segundo tema, para esto se introduce al estudiante en una mitología que le permita desarrollar competencias para identificar datos y procesos, mismos que son formalizados en el tercer tema, en la cual se presentan métodos formales y generalizados para representar la solución a problemas.

Para poder resolver problemas computacionalmente, es necesario saber cómo son representados los datos en una computadora, y como es el esqueleto de un programa que pueda utilizar dichos datos para resolver un problema, estos conocimientos son desarrollados por el estudiante en el cuarto tema.

En el quinto tema es cuando el estudiante ya debe ser capaz de escribir programas, poder darle instrucciones a la computadora que esta nos pida y nos reporte datos. En este tema también se introducen los principios de programación que permiten al programador describir toma de decisiones y procesos iterativos.

En las actividades científico-tecnológicas siempre es necesario el uso de vectores, el sexto tema presenta la versión computacional de este concepto, induciendo al estudiante para la elección correcta de estructuras que permiten la organización y procesamiento de información, sobre todo cuando esta viene en grandes volúmenes.

En el proceso natural de solución de problemas, generalmente optamos por hacer procesos pequeños a la vez, en lugar de hacer procesos largos y tediosos, este principio es incluido en el séptimo tema, donde además se presentan los conceptos y técnicas necesarios para la programación de funciones, permitiendo al estudiante implementar computacionalmente este concepto.

Es común que en la solución (sobre todo numérica) de problemas, se requiera de procesos que requieren del almacenamiento de grandes cantidades de información, misma que tiene una vigencia temporal puesto que es actualizada constantemente o simplemente desaparece o se vuelve innecesaria después de ciertos pasos o tiempo. Una metodología para optimizar el uso de memoria RAM para estos casos, es presentado en el octavo tema.

El profesor debe ser conocedor de la disciplina que está bajo su responsabilidad, conocer su origen y desarrollo histórico para considerar este conocimiento al abordar los temas. Desarrollar la capacidad para coordinar y trabajar en equipo; orientar el trabajo del estudiante y potenciar en él la autonomía, el trabajo cooperativo y la toma de decisiones.

**3. Participantes en el diseño y seguimiento curricular del programa**

<b>Lugar y fecha de elaboración o revisión</b>	<b>Participantes</b>	<b>Evento</b>
<p>Instituto Tecnológico de Ciudad Juárez del 27 al 29 de Abril de 2009.</p>	<p>Representantes de los Institutos Tecnológicos de: Tijuana, Querétaro, Celaya, Saltillo, Ciudad Juárez, Superior de Irapuato, San Luis Potosí, Chihuahua.</p>	<p>Reunión Nacional de diseño e innovación curricular para el desarrollo de competencias profesionales de las carreras de Ingeniería en Nanotecnología e Ingeniería en Logística del SNEST.</p>
<p>Instituto Tecnológico de Puebla del 8 al 12 de Junio de 2009.</p>	<p>Representantes de los Institutos Tecnológicos de: Tijuana, Querétaro, Celaya, Saltillo, Ciudad Juárez, Superior de Irapuato, San Luis Potosí, Chihuahua.</p>	<p>Reunión de seguimiento de diseño e innovación curricular para el desarrollo de competencias profesionales de las carreras de Ing. en Nanotecnología, Gestión Empresarial, Logística, y asignaturas comunes del SNEST.</p>
<p>Instituto Tecnológico de Mazatlán del 23 al 27 de Noviembre de 2009.</p>	<p>Representantes de los Institutos Tecnológicos de: Tijuana, Querétaro, Ciudad Juárez, Superior de Irapuato, San Luis Potosí, Chihuahua.</p>	<p>Reunión de seguimiento de diseño e innovación curricular para el desarrollo de competencias profesionales de la carrera de Ing. en Nanotecnología, del SNEST.</p>
<p>Instituto Tecnológico de Villahermosa del 24 al 28 de Mayo de 2010.</p>	<p>Representantes de los Institutos Tecnológicos de: Tijuana, Querétaro, Superior de Irapuato, Chihuahua, Saltillo.</p>	<p>Reunión de consolidación de diseño e innovación curricular para el desarrollo de competencias profesionales de la carrera de Ing. en Nanotecnología, del SNEST.</p>
<p>Tecnológico Nacional de México, del 26 al 30 de agosto de 2013.</p>	<p>Representantes de los Institutos Tecnológicos de: Boca del Río y Mazatlán.</p>	<p>Reunión Nacional de Seguimiento Curricular de las carreras de Ingeniería en Nanotecnología, Ingeniería Petrolera, Ingeniería en Acuicultura, Ingeniería en Pesquerías, Ingeniería Naval y Gastronomía del SNIT.</p>

**4. Competencia(s) a desarrollar**

<b>Competencia(s) específica(s) de la asignatura</b>
<ul style="list-style-type: none"> <li>Plantea metodológicamente la solución de problemas susceptibles de ser computarizados a través del manejo de técnicas estructuradas de diseño y formulación de algoritmos, así como realizar la documentación de manera pertinente.</li> </ul>

**5. Competencias previas**

<ul style="list-style-type: none"> <li>Utiliza equipo de cómputo y software básico.</li> <li>Aplica los conocimientos básicos de Matemáticas (Álgebra, Trigonometría y Geometría Analítica).</li> </ul>
---

**6. Temario**

No.	Temas	Subtemas
1	Conceptos básicos	1.1 Introducción 1.2 De los problemas a los programas 1.3 Definición de lenguaje 1.4 Definición de algoritmo 1.5 Algoritmos cotidianos 1.6 Definición de lenguajes algorítmicos
2	Metodología para la solución de problemas por medio de computadoras	2.1 Análisis del problema 2.2 Diseño del algoritmo 2.3 Codificación 2.4 Compilación y ejecución 2.5 Verificación y depuración 2.6 Mantenimiento y documentación
3	Técnicas para la formulación de algoritmos	3.1 Diagramas de flujo 3.2 Pseudocódigo
4	Principios de programación	4.1 Tipos de datos 4.2 Estructura básica de un programa
5	Estructuras Básicas de control y Arreglos	5.1 Secuenciales 5.2 Asignación 5.3 Entrada 5.4 Salida 5.5 Condicionales 5.5.1 Simple 5.5.2 Múltiple 5.6 Estructuras de repetición 5.6.1 Por contador 5.6.2 Por condición 5.7 Arreglos 5.7.1 Arreglos unidimensionales 5.7.2 Arreglos multidimensionales

**7. Actividades de aprendizaje de los temas**

<b>1. Conceptos básicos</b>	
<b>Competencias</b>	<b>Actividades de aprendizaje</b>
<p>Específica(s):</p> <ul style="list-style-type: none"> <li>• Comprende los conceptos básicos de programación utilizados en la solución de problemas por medio de la computadora.</li> </ul> <p>Genéricas:</p> <ul style="list-style-type: none"> <li>• Capacidad de análisis y síntesis</li> <li>• Capacidad de organizar y planificar</li> <li>• Habilidades básicas de manejo de la computadora</li> <li>• Habilidad para buscar y analizar información proveniente de fuentes diversas</li> <li>• Solución de problemas</li> <li>• Toma de decisiones.</li> <li>• Trabajo en equipo</li> <li>• Capacidad de aprender</li> <li>• Capacidad de generar nuevas ideas (creatividad)</li> <li>• Habilidad para trabajar en forma autónoma.</li> </ul>	<ul style="list-style-type: none"> <li>• Comprender el significado y concepto de algoritmo.</li> <li>• Desarrollar y aplica el concepto de algoritmo en problemas reales aplicados en su entorno.</li> </ul>
<b>2. Metodología para la solución de problemas por medio de computadoras</b>	
<b>Competencias</b>	<b>Actividades de aprendizaje</b>
<p>Específica(s):</p> <ul style="list-style-type: none"> <li>• Aplica de manera secuencial los pasos para proponer soluciones a problemas por medio de computadora.</li> </ul> <p>Genéricas:</p> <ul style="list-style-type: none"> <li>• Capacidad de análisis y síntesis</li> <li>• Capacidad de organizar y planificar</li> <li>• Habilidades básicas de manejo de la computadora</li> <li>• Habilidad para buscar y analizar información proveniente de fuentes diversas</li> <li>• Solución de problemas</li> <li>• Toma de decisiones.</li> </ul>	<ul style="list-style-type: none"> <li>• Visualizar algún problema preferentemente del área de nanotecnología y metodológicamente diseñar una solución.</li> </ul>

<ul style="list-style-type: none"> <li>• Trabajo en equipo</li> <li>• Capacidad de aprender</li> <li>• Capacidad de generar nuevas ideas (creatividad)</li> <li>• Habilidad para trabajar en forma autónoma.</li> </ul>	
<b>3. Técnicas para la formulación de algoritmos</b>	
<b>Competencias</b>	<b>Actividades de aprendizaje</b>
<p>Específica(s):</p> <ul style="list-style-type: none"> <li>• Selecciona y utiliza de manera correcta técnicas para la representación y formulación de algoritmos.</li> </ul> <p>Genéricas:</p> <ul style="list-style-type: none"> <li>• Capacidad de análisis y síntesis</li> <li>• Capacidad de organizar y planificar</li> <li>• Habilidades básicas de manejo de la computadora</li> <li>• Habilidad para buscar y analizar información proveniente de fuentes diversas</li> <li>• Solución de problemas</li> <li>• Toma de decisiones.</li> <li>• Trabajo en equipo</li> <li>• Capacidad de aprender</li> <li>• Capacidad de generar nuevas ideas (creatividad)</li> <li>• Habilidad para trabajar en forma autónoma.</li> </ul>	<ul style="list-style-type: none"> <li>• Trabajar en grupos para discutir sobre la técnica a seguir para representar y formular un algoritmo para un problema en específico.</li> <li>• Diseñar algoritmos de casos específicos usando la secuencia de diagramas de flujo y pseudocódigos.</li> </ul>
<b>4. Principios de programación</b>	
<b>Competencias</b>	<b>Actividades de aprendizaje</b>
<p>Específica(s):</p> <ul style="list-style-type: none"> <li>• Traduce algoritmos a un lenguaje de programación para la resolución de problemas.</li> </ul> <p>Genéricas:</p> <ul style="list-style-type: none"> <li>• Capacidad de análisis y síntesis</li> <li>• Capacidad de organizar y planificar</li> <li>• Habilidades básicas de manejo de la computadora</li> <li>• Habilidad para buscar y analizar información proveniente de fuentes diversas</li> </ul>	<ul style="list-style-type: none"> <li>• Utilizar algoritmos desarrollados en las unidades 2 y 3 para identificar los diferentes tipos de datos.</li> <li>• Programar estos algoritmos en un lenguaje de programación de nivel alto.</li> </ul>

<ul style="list-style-type: none"> <li>• Solución de problemas</li> <li>• Toma de decisiones.</li> <li>• Trabajo en equipo</li> <li>• Capacidad de aprender</li> <li>• Capacidad de generar nuevas ideas (creatividad)</li> <li>• Habilidad para trabajar en forma autónoma.</li> </ul>	
<b>5. Estructuras Básicas de control</b>	
<b>Competencias</b>	<b>Actividades de aprendizaje</b>
<p>Específica(s):</p> <ul style="list-style-type: none"> <li>• Elabora programas orientados a la aplicación de las diferentes estructuras de control.</li> <li>• Aplica arreglos en programas para resolver problemas que requieran este tipo de estructura.</li> </ul> <p>Genéricas:</p> <ul style="list-style-type: none"> <li>• Capacidad de análisis y síntesis</li> <li>• Capacidad de organizar y planificar</li> <li>• Habilidades básicas de manejo de la computadora</li> <li>• Habilidad para buscar y analizar información proveniente de fuentes diversas</li> <li>• Solución de problemas</li> <li>• Toma de decisiones.</li> <li>• Trabajo en equipo</li> <li>• Capacidad de aprender</li> <li>• Capacidad de generar nuevas ideas (creatividad)</li> <li>• Habilidad para trabajar en forma autónoma.</li> </ul>	<ul style="list-style-type: none"> <li>• Resolver problemas a través de programas con diferentes estructuras y comentar las diferencias.</li> <li>• Busca aplicaciones similares a las expuestas por el profesor para su exposición.</li> <li>• Discutir en la clase la solución a estos problemas.</li> <li>• Elabora algoritmos y programas para la solución de problemas que involucren arreglos de una o mas dimensiones.</li> </ul>

## 8. Práctica(s)

- Desarrollar y aplicar el concepto de algoritmo en problemas reales aplicados en su entorno.
- Desarrollar y aplicar el concepto de algoritmo y la metodología en problemas orientados a su área de conocimientos.
- Elaborar diagramas de flujo y programas para la solución de problemas orientados a su área de conocimiento utilizando un lenguaje de programación.
- Diseñar y programar la solución de problemas prácticos utilizando las estructuras de control.
- Elaborar algoritmos y programas para la solución de problemas que involucren arreglos de una o más dimensiones.
- Elaborar programas sencillos, pero aplicando el concepto de módulos que hace uso de un "menú" principal.
- Elaborar programas para generar una lista simplemente ligada.

## 9. Proyecto de asignatura

El objetivo del proyecto que planteé el docente que imparta esta asignatura, es demostrar el desarrollo y alcance de la(s) competencia(s) de la asignatura, considerando las siguientes fases:

- **Fundamentación:** marco referencial (teórico, conceptual, contextual, legal) en el cual se fundamenta el proyecto de acuerdo con un diagnóstico realizado, mismo que permite a los estudiantes lograr la comprensión de la realidad o situación objeto de estudio para definir un proceso de intervención o hacer el diseño de un modelo.
- **Planeación:** con base en el diagnóstico en esta fase se realiza el diseño del proyecto por parte de los estudiantes con asesoría del docente; implica planificar un proceso: de intervención empresarial, social o comunitario, el diseño de un modelo, entre otros, según el tipo de proyecto, las actividades a realizar los recursos requeridos y el cronograma de trabajo.
- **Ejecución:** consiste en el desarrollo de la planeación del proyecto realizada por parte de los estudiantes con asesoría del docente, es decir en la intervención (social, empresarial), o construcción del modelo propuesto según el tipo de proyecto, es la fase de mayor duración que implica el desempeño de las competencias genéricas y específicas a desarrollar.



- **Evaluación:** es la fase final que aplica un juicio de valor en el contexto laboral-profesión, social e investigativo, ésta se debe realizar a través del reconocimiento de logros y aspectos a mejorar se estará promoviendo el concepto de “evaluación para la mejora continua”, la metacognición, el desarrollo del pensamiento crítico y reflexivo en los estudiantes.

## 10. Evaluación por competencias

La evaluación debe ser continua y cotidiana e imparcial, por lo que se debe considerar el desempeño en cada una de las actividades de aprendizaje, así como todo su contexto: autoevaluación, coevaluación y heteroevaluación.

- Autoevaluación (puede ser mediante un cuestionario breve)
- Desempeño conceptual
- Desempeño actitudinal y valores de responsabilidad
- Desempeño en habilidades
- Coevaluación (puede ser mediante un cuestionario breve en donde los estudiantes evalúen a sus compañeros)
- En desempeño actitudinal y valores en trabajo colaborativo
- Heteroevaluación
- Diagnóstica
- Desempeño conceptual
- Desempeño en habilidades
- Desempeño actitudinal y valores
- Reportes de las observaciones hechas durante las actividades, así como de las conclusiones obtenidas de dichas observaciones.
- Reportes de investigación documental que contengan los requisitos mínimos acordados en la clase.
- Exámenes escritos para comprobar el manejo de aspectos teóricos y declarativos y que no determine más del 60% de la calificación.

## 11. Fuentes de información

1. Aho, Hopcroft, Ullman, *Estructura de Datos y Algoritmos*, Ed. Sitesa.
2. Luis Joyanes Aguilar, *Fundamentos de Programación, Algoritmos y Estructura de Datos*.
3. Guillermo Levine, *Introducción a la Computación y la Programación Estructurada*, Ed. Mc. Graw Hill.
4. Goodman. Hedetmienic, *Introduction to the Design and Analysis of Algorithms*, Ed. Mc. Graw Hill.
5. Francisc Scheid, *Computers and Programing*, Ed. Mc. Graw Hill.
6. Tremblay Sorensen, *An Introduction to Data Structures with Applications*, Ed. Mc. Graw Hill.
7. Deitel, Java: *How to Program*, Pearson.
8. Deitel, C#: *How to Program*, Pearson.