



### 1. Datos Generales de la asignatura

<b>Nombre de la asignatura:</b>	Programación Estructurada
<b>Clave de la asignatura:</b>	SED-2319
<b>SATCA<sup>1</sup>:</b>	2-3-5
<b>Carrera:</b>	Ingeniería en Semiconductores

### 2. Presentación

<b>Caracterización de la asignatura</b>
<p>Esta asignatura aporta al perfil profesional la capacidad para desarrollar aplicaciones en un lenguaje de programación de alto nivel, para la solución de problemas relacionados con las diferentes disciplinas en el área de Ingeniería en Semiconductores.</p> <p>Las competencias adquiridas en asignatura permitirán al estudiante desarrollar aplicaciones de cómputo en el modo de consola, utilizando un lenguaje de programación de alto nivel orientado al cálculo numérico, las cuales facilitarán la solución de problemas de su área, tales como los fenómenos físicos cubiertos en la asignatura de mecánica clásica, circuitos de corriente directa estudiados en la asignatura de electromagnetismo, y de estadística descriptiva definida en la materia de probabilidad y estadística.</p> <p>El desarrollo de programas estructurados aporta las bases de programación para la solución de problemas de cálculo en ingeniería y para el desarrollo de programas de aplicación con interfaces gráficas de usuario. Es un recurso valioso que permite la implementación de los algoritmos de la materia de análisis numérico.</p> <p>La importancia de esta materia se encuentra en los fundamentos de programación, así como el uso de puertos de comunicación que permiten el desarrollo de programas de aplicación, utilizando un lenguaje de programación de alto nivel para su aplicación en la solución de problemas propios de la ingeniería.</p> <p>También, se relaciona con la materia de Programación Estructurada, en los temas de elementos básicos de las interfaces gráficas de usuario, desarrollo de interfaces gráficas de usuario y el desarrollo de aplicaciones.</p>
<b>Intención didáctica</b>
<p>La asignatura se compone por seis temas; el primero aborda las bases para el desarrollo de la lógica de programación de computadoras para la resolución de problemas de ingeniería. Se sugiere que el docente presente al estudiante algunos programas de aplicación utilizados en diferentes áreas. Para el diseño y la simulación de algoritmos se sugiere abordar problemas del área de las ciencias básicas de la ingeniería como física, electromagnetismo, álgebra, aritmética y probabilidad y estadística.</p>

<sup>1</sup> Sistema de Asignación y Transferencia de Créditos Académicos



El segundo tema se destina al estudio de los elementos propios del lenguaje de programación. La academia correspondiente deberá seleccionar el lenguaje de programación orientado al cálculo numérico más adecuado a su entorno laboral, así como la plataforma de desarrollo por utilizar. Se recomienda utilizar el lenguaje de programación Python por su amplio uso en el desarrollo de software para pruebas de funcionamiento en circuitos integrados.

En el tercer tema se proporcionan los conocimientos sobre estructuras de control, que permitirán el desarrollo de programas complejos. Se recomienda un enfoque progresivo, en el cual se vayan desarrollando aplicaciones simples que demuestren el uso adecuado de cada uno de los elementos de control y a medida que se avanza en el programa, construir aplicaciones complejas y funcionales, integrando las diferentes estructuras en un solo programa.

En el cuarto tema se estudian los arreglos unidimensionales (vectores) y bidimensionales (matrices). Es importante que el estudiante aprenda a desplazarse a través de los arreglos de tal manera que le sea posible leer, escribir y modificar cada uno de los elementos de éstos. Se sugiere que se refuercen los contenidos con ejemplos propios del área.

El quinto tema aborda la metodología de funciones de código basada en la programación modular que permite una mayor flexibilidad tanto en su reutilización como en su mantenimiento. Se sugiere el uso de bibliotecas de funciones que permitan la entrada y salida de datos, su almacenamiento permanente en archivos y el tratamiento de datos. Se recomienda desarrollar programas que integren múltiples alternativas de operación mediante el uso de menús y algoritmos recursivos haciendo uso de funciones creadas por el estudiante y funciones existentes en las bibliotecas.

En el sexto tema se estudian los diferentes puertos de comunicación en una computadora. Se realizan comunicaciones de la computadora con el mundo exterior. En el tema, los puertos de comunicación se dividen en dos categorías: puertos paralelos y puertos seriales. Algunos protocolos propuestos son: RS-232, USB, Bluetooth y WiFi.

El docente que imparta esta asignatura debe tener amplia experiencia profesional en elementos de lenguaje de programación y el uso de puerto de comunicación, además debe ser un profesor con las competencias docentes necesarias que le permitan optimizar las herramientas en forma adecuada para facilitar el aprendizaje y motivar a los estudiantes a obtener y aplicar los conocimientos, además debe fomentar el trabajo en equipo, el aprendizaje autónomo y el aprendizaje basado en proyectos o en problemas.

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

Lugar y fecha de elaboración o revisión	Participantes	Observaciones
Tecnológico Nacional de México, del 24 al 28 de abril de 2023.	Representantes de los Institutos Tecnológicos de: Aguascalientes, Ciudad Madero, Irapuato,	Reunión Nacional de Diseño e Innovación Curricular para el Desarrollo y Formación de Competencias Profesionales de la



	Matamoros, Hermosillo y Purísima del Rincón.	Carrera de Ingeniería en Semiconductores.
Tecnológico Nacional de México, del 22 al 24 de mayo de 2023.	Representantes de los Institutos Tecnológicos de: Aguascalientes, Ciudad Madero, Irapuato, Matamoros, Hermosillo y Purísima del Rincón.	Reunión Nacional de Consolidación Curricular para el Desarrollo y Formación de Competencias Profesionales de la Carrera de Ingeniería en Semiconductores.

#### 4. Logro formativo a desarrollar en la asignatura

##### Saberes, habilidades y destrezas de la asignatura

Diseña algoritmos y desarrolla programas de aplicación en modo de consola, utilizando un lenguaje de programación de alto nivel mediante el paradigma de la programación estructurada, para su aplicación en la solución de problemas propios de la ingeniería.

#### 5. Saberes, habilidades y destrezas previas

- Uso de computadora personal.
- Capacidad de abstracción, análisis y síntesis.
- Habilidades en el uso de las tecnologías de la información y de la comunicación.
- Conoce y aplica los conceptos de álgebra y aritmética para la resolución de problemas matemáticos.
- Conoce y aplica los conceptos de probabilidad y estadística para resolver problemas de ingeniería.

#### 6. Temario

No	Temas	Subtemas
1	Fundamentos de programación	1.1. Clasificación de los lenguajes de programación. 1.2. Diseño de algoritmos 1.2.1. Pseudocódigo. 1.2.2. Diagramas de flujo. 1.2.3. Uso de programas de simulación de pseudocódigo y diagramas de flujo. 1.3. Compiladores en lenguajes de alto nivel.
2	Elementos del lenguaje de programación	2.1. Introducción al entorno de programación. 2.2. Estructura básica de un programa. 2.2.1. Comentarios. 2.2.2. Identificadores. 2.2.3. Palabras reservadas. 2.2.4. Clases. 2.2.5. Tipos de datos. 2.2.5.1. Simples.



		<p>2.2.5.2. Compuestos.</p> <p>2.2.6. Variables y Constantes.</p> <p>2.2.7. Atributos.</p> <p>2.2.8. Operadores.</p> <p>2.2.8.1. Aritméticos.</p> <p>2.2.8.2. Lógicos.</p> <p>2.2.8.3. De comparación.</p> <p>2.2.8.4. De desplazamiento.</p> <p>2.2.9. Manejo de cadena de caracteres.</p>
3	Estructuras de control	<p>3.1. Estructuras de condición</p> <p>3.1.1. IF</p> <p>3.1.2. IF/ELSE</p> <p>3.1.3. IF/ELSIF</p> <p>3.2. Estructuras de repetición.</p> <p>3.2.1. WHILE</p> <p>3.2.2. WHILE</p> <p>3.2.3. FOR</p> <p>3.3. Estructura de múltiple selección.</p> <p>3.3.1. SWITCH. /CASE</p> <p>3.4. Formulación y aplicación de algoritmos</p> <p>3.5. utilizando estructuras de control.</p>
4	Arreglos	<p>4.1. Definición e importancia de los arreglos en la programación.</p> <p>4.2. Declaración de arreglos.</p> <p>4.2.1. Vectores.</p> <p>4.2.2. Matrices.</p> <p>4.3. Lectura y escritura de arreglos.</p> <p>4.3.1. Vectores.</p> <p>4.3.2. Matrices.</p> <p>4.4. Operaciones con arreglos.</p>
5	Funciones	<p>5.1. Estructura de la función.</p> <p>5.2. Llamado o invocación de una función.</p> <p>5.3. Uso de funciones con parámetros.</p> <p>5.3.1. De entrada.</p> <p>5.3.2. De salida.</p> <p>5.4. Funciones externas.</p> <p>5.4.1. Del usuario.</p> <p>5.4.2. De bibliotecas.</p>
6	Uso de puertos de comunicación	<p>6.1. Tipos de puertos de comunicación.</p> <p>6.2. Especificaciones de los puertos de comunicación serie y paralelo.</p>



		6.2.1. Lectura y escritura de datos a través de un puerto de comunicación (RS-232, USB, Bluetooth o WiFi)
--	--	---

## 7. Actividades de aprendizaje de los temas

<b>1. Fundamentos de programación</b>	
Saberes, habilidades y destrezas	Actividades de aprendizaje
<p>Conoce la clasificación de los lenguajes, dando realce a la programación de alto nivel para desarrollar y simular algoritmos y diagramas de flujo agilizando el pensamiento lógico en la solución de problemas de ingeniería orientados a su campo laboral.</p> <ul style="list-style-type: none"> <li>• Capacidad cognitiva para comprender y manipular ideas e información.</li> <li>• Capacidad de comunicación oral y escrita.</li> </ul>	<ul style="list-style-type: none"> <li>• Investigar en diferentes fuentes de información los lenguajes de programación y su clasificación.</li> <li>• Exponer frente a grupo los resultados de la investigación realizada.</li> <li>• Desarrollar y simular algoritmos para solucionar problemas delimitados usando pseudocódigo y diagramas de flujo. Algunas herramientas para realizar estas actividades son: PSeInt, Rapto, DFD, Visustin y Code2flow</li> <li>• Investigar la importancia del compilador en los lenguajes de alto nivel.</li> <li>• Investigar los lenguajes orientados al cálculo numérico que apoyan al campo de la ingeniería.</li> </ul>
<b>2. Elementos del lenguaje de programación</b>	
Saberes, habilidades y destrezas	Actividades de aprendizaje
<p>Conoce el entorno de programación y utiliza los elementos básicos del lenguaje de programación, desarrollando programas con aplicaciones básicas para la solución de problemas reales.</p> <ul style="list-style-type: none"> <li>• Capacidad cognitiva para comprender y manipular ideas e información.</li> <li>• Trabajo en equipo.</li> </ul>	<ul style="list-style-type: none"> <li>• Familiarizarse con el entorno de programación por medio del diseño, edición y compilación de programas sencillos en un lenguaje de programación orientado al cálculo numérico.</li> <li>• Investigar las palabras reservadas del lenguaje de programación utilizado.</li> <li>• Desarrollar programas que utilicen variables, constantes y los diferentes tipos de datos.</li> <li>• Desarrollar programas que utilicen operadores aritméticos, lógicos y de comparación.</li> <li>• Desarrollar programas que utilicen la mayoría de los puntos vistos en la estructura básica de un programa con la finalidad de dar solución a problemas cotidianos.</li> </ul>
<b>3. Estructuras de control.</b>	
Saberes, habilidades y destrezas	Actividades de aprendizaje
<p>Desarrolla programas a partir del diseño de algoritmos utilizando estructuras de control, para la solución de problemas reales en ingeniería aplicados a la industria.</p>	<ul style="list-style-type: none"> <li>• Realizar programas básicos para cada una de las estructuras de selección, repetición y múltiple selección.</li> <li>• Realizar programas complejos para solucionar problemas reales utilizando las estructuras de</li> </ul>



<ul style="list-style-type: none"> <li>• Capacidad creativa.</li> <li>• Capacidad cognitiva para comprender y manipular ideas e información.</li> <li>• Capacidad para tomar decisiones.</li> <li>• Capacidad de comunicación oral y escrita.</li> <li>• Trabajo en equipo.</li> </ul>	<p>control necesarias en cada uno de los programas.</p> <ul style="list-style-type: none"> <li>• Proponer soluciones alternativas para los programas realizados.</li> </ul>
<b>4. Arreglos</b>	
Saberes, habilidades y destrezas	Actividades de aprendizaje
<p>Aplica arreglos unidimensionales y multidimensionales para la solución de problemas reales en ingeniería orientados a su campo laboral.</p> <ul style="list-style-type: none"> <li>• Capacidad creativa.</li> <li>• Capacidad cognitiva para comprender y manipular ideas e información.</li> <li>• Trabajo en equipo.</li> <li>• Capacidad para tomar decisiones.</li> <li>• Capacidad de comunicación oral y escrita.</li> </ul>	<ul style="list-style-type: none"> <li>• Investigar el uso de arreglos en aplicaciones reales de ingeniería, así como en las asignaturas de semestres posteriores.</li> <li>• Desarrollar programas que utilicen arreglos unidimensionales y multidimensionales con ejemplos didácticos para la comprensión de la lectura, escritura y modificación de datos en dichos arreglos.</li> <li>• Desarrollar programas con arreglos que permitan operaciones entre ellos.</li> </ul>
<b>5. Funciones</b>	
Saberes, habilidades y destrezas	Actividades de aprendizaje
<p>Desarrolla programas estructurados utilizando funciones creadas por el estudiante y funciones de las bibliotecas del lenguaje de programación utilizado, para la resolución de problemas reales.</p> <ul style="list-style-type: none"> <li>• Identifica las ventajas y desventajas de la programación lineal y la programación estructurada.</li> <li>• Capacidad creativa.</li> <li>• Capacidad cognitiva para comprender y manipular ideas e información.</li> </ul>	<ul style="list-style-type: none"> <li>• Investigar los elementos que conforman una función, método o subfunción, dependiendo del lenguaje de programación utilizado.</li> <li>• Rediseñar los programas elaborados previamente y estructurarlos utilizando funciones creadas por el estudiante.</li> <li>• Desarrollar programas utilizando las funciones existentes en las bibliotecas del lenguaje de programación para el manejo de archivos, manipulación de cadenas de texto, entrada y salida de datos, funciones matemáticas y de conversión de tipo de datos.</li> </ul>



<ul style="list-style-type: none"> <li>• Capacidad para tomar decisiones.</li> <li>• Trabajo en equipo.</li> <li>• Capacidad de comunicación oral y escrita.</li> </ul>	
<b>6. Uso de puertos de comunicación</b>	
Saberes, habilidades y destrezas	Actividades de aprendizaje
<p>Desarrolla programas utilizando los puertos de la computadora para interactuar con sistemas electrónicos externos aplicados en la solución de problemas de ingeniería dentro de su entorno profesional.</p> <ul style="list-style-type: none"> <li>• Capacidad creativa.</li> <li>• Capacidad cognitiva para comprender y manipular ideas e información.</li> <li>• Trabajo en equipo.</li> <li>• Capacidad para tomar decisiones.</li> <li>• Capacidad de investigación.</li> <li>• Capacidad de comunicación oral y escrita.</li> </ul>	<ul style="list-style-type: none"> <li>• Investigar y exponer frente a grupo los diferentes tipos de puertos de comunicación alámbricos e inalámbricos, seriales y paralelos.</li> <li>• Comprender la interacción de los puertos de comunicación con los sistemas electrónicos.</li> <li>• Desarrollar programas que permitan interactuar con sistemas electrónicos externos a la computadora (motores, led, actuadores, sensores, microcontroladores, etc.)</li> </ul>

### 8. Práctica(s)

<ul style="list-style-type: none"> <li>• Desarrollar y simular algoritmos (pseudocódigo y diagramas de flujo) utilizando una herramienta computacional como PSeInt, Code2Flow, Visustin, entre otras.</li> <li>• Desarrollar un programa que muestre un mensaje en la consola.</li> <li>• Desarrollar programas que almacenen un valor en una variable y realice operaciones aritméticas con constantes y muestre el resultado en pantalla.</li> <li>• Desarrollar programas que utilicen variables creadas por el estudiante y constantes del lenguaje de programación como el valor de pi, por ejemplo, para calcular el área de un círculo asignando el valor del radio y posteriormente ingresando el valor del radio por el usuario desde el teclado.</li> <li>• Desarrollar un programa que realice la conversión entre sistemas numéricos.</li> <li>• Desarrollar un programa que, a partir del ingreso de valores binarios de 8 bit, realice las operaciones lógicas AND, OR, XOR y XNOR entre los valores ingresados.</li> <li>• Desarrollar programas en el que se generen funciones que realicen las operaciones aritméticas básicas entre dos números dados por el usuario.</li> <li>• Desarrollar un programa que utilice una función que calcule la factorial de un número.</li> <li>• Desarrollar un programa que a partir de un rango de años obtenga los que son bisiestos.</li> <li>• Desarrollar un programa que genere números de la serie fibonacci y se almacenen en un archivo.</li> </ul>
---



- Desarrollar programas que utilicen arreglos para almacenar números ingresados por el usuario. Posteriormente realizar operaciones matemáticas con estos datos, como suma, multiplicación, división, promedio, detección de número mayor, detección de número menor.
- Desarrollar programas que muestren un menú para realizar operaciones con vectores y matrices, como matriz inversa, matriz cuadrada, suma de matrices, multiplicación de matrices, multiplicación de matrices por un escalar, multiplicación de elemento por elemento, ordenar valores de las matrices, entre otras.
- Desarrollar programas que manipulen datos de una cadena como arreglo de caracteres para realizar separación de caracteres, concatenación, conversión de tipo de datos.
- Desarrollar programas que utilicen las funciones de las bibliotecas del lenguaje de programación para realizar operaciones matemáticas como redondeo hacia arriba y hacia abajo, truncamiento, conversión de tipos de datos numéricos, raíces y potencias, promedio, funciones estadísticas, entre otras.
- Desarrollar programas que utilicen funciones de las bibliotecas del lenguaje de programación para manipular cadenas de texto.
- Desarrollar programas que resuelvan problemas de física.
- Desarrollar programas que resuelvan problemas de circuitos eléctricos de corriente directa (Ley de Ohm, leyes de Kirchhoff [sistemas de ecuaciones lineales], resistores en serie y paralelo)
- Desarrollar un programa que comunique la computadora con el exterior para realizar una animación gráfica con leds.
- Desarrollar un programa que permita a la computadora leer datos desde un sensor externo y que almacene los datos en un archivo en formato de tabla.

## 9. Proyecto de asignatura

El objetivo del proyecto que planteé el docente que imparta esta asignatura, es demostrar el desarrollo y alcance del(los) logro(s) formativo(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 los saberes, habilidades y destrezas 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”, el desarrollo del pensamiento crítico y reflexivo en los estudiantes.

## 10. Evaluación de saberes, habilidades y destrezas

Debe aplicarse evaluación:

- Diagnóstica, al inicio del curso, sin que se considere para la evaluación sumativa.
- Formativa, durante el desarrollo de la asignatura, apoyándose en los instrumentos y herramientas que se señalan a continuación.
- Sumativa, al final, para determinar la valoración numérica de la asignatura se debe basar en los niveles de desempeño establecidos en el Lineamiento para la Evaluación y Acreditación de Asignaturas vigente.

Se recomienda el uso de la coevaluación, autoevaluación y heteroevaluación.

Todos los productos deben de estar contenidos en el portafolio de evidencias que el alumno integrará durante el desarrollo de la asignatura. El docente tendrá en resguardo dicho portafolio al finalizar el curso.

El portafolio de evidencias puede ser electrónico.

Instrumentos y herramientas sugeridas para evaluar las actividades de aprendizaje:

- Mapas conceptuales.
- Cuestionarios teóricos/prácticos.
- Representaciones esquemáticas y gráficas.
- Mapas mentales.
- Reportes de prácticas y simulaciones.
- Resúmenes.
- Rúbricas.
- Listas de cotejo.
- Matriz de valoración.
- Guía de observación.

## 11. Referencias

1. Downey, A. B. (2015). *Think Python: How to Think Like a Computer Scientist*. O'Reilly Media.
2. Severance, C.R. (2020). *Python para todos: Explorando datos en Python 3*. Publicado independientemente en la plataforma CreateSpace..
3. Chazallet, S. (2020). *Python 3. Los fundamentos del lenguaje*. Eni.
4. Libros de programación con python.