

**Asignatura** Fundamentos de la Programación**Número** COP2210**Créditos**

3

**Horas**

45

**Horas asignadas fuera de clase**

90

**Periodo Académico**

Por determinar

**Prerrequisitos**

—

**Correquisitos**

—

**Horario y lugar de la asignatura**

Es un programa online y el estudio se basa en materiales disponibles de forma asincrónica. Las únicas actividades presenciales son los webinars, planeados por el profesor. A continuación, se presentan más detalles sobre los horarios del programa y de la asignatura.

**Política de asistencia**

La asistencia e implicación por parte del estudiante se mide por la participación en actividades académicas y de evaluación. Por lo tanto, participación en los foros semanales es obligatoria para aprobar la asignatura. Acceder al aula virtual o mandar mensajes a través del email del campus no cuentan como participación en la asignatura.

**Política de tardanza**

Trabajos o entradas en los foros entregados fuera de plazo no se considerarán y recibirán una nota de suspenso (F).

NOTA: El plagio se define como el uso, sin el debido reconocimiento, de las ideas, frases, o mayores unidades del discurso de otro escritor u orador. El plagio incluye la copia de software sin autorización y la violación de las leyes de derechos de autor. Estudiantes que plagian recibirán una nota de suspenso en su examen o trabajo.

**Nombre y contacto del profesor**

Por determinar.

Horario de oficina:

El profesor está disponible fuera de las horas de clase para responder a cualquier pregunta o inquietud relacionada con este curso. Los estudiantes pueden ponerse en contacto con su profesor las 24 horas del día, los 7 días de la semana a través del foro Pregúntale al Profesor en Canvas LMS. Los profesores responderán a todas las consultas en un plazo de 48 horas.

## Libros y materiales obligatorios

La institución reconoce el uso de libros de texto en el aula como parte de su metodología académica. El libro de texto forma parte del plan de estudio y se utiliza para comunicar con los estudiantes en el aula de forma eficaz. La universidad aportará el material necesario para la asignatura.

## Responsabilidades del estudiante

### Descripción de la asignatura

El objetivo de esta asignatura es el de formar a los estudiantes como desarrolladores de nivel inicial. Estudiantes iniciales aprenderán los principios básicos de la programación informática, utilizando un idioma de programación específico, orientado hacia el objeto. Después, los estudiantes aprenderán el proceso de desarrollo. Trabajarán en proyectos individuales que reflejan escenarios reales del sector.

### Competencias de la asignatura

Al final de esta asignatura, el estudiante podrá:

- ▶ Comprender los principios básicos de programación orientada a objetos y el diseño de algoritmos.
- ▶ Analizar la estructura de un programa y modificar el flujo de ejecución de las instrucciones.
- ▶ Diseñar y desarrollar programas que resuelvan problemas concretos bajo el paradigma de la programación orientada a objetos.

### Horario de la Asignatura:

SEMANA	CONTENIDO	
Semana 1	Objetivos específicos	<p>Aquí se realiza una aproximación desde el punto de vista teórico a los lenguajes de programación, la estructura básica de un computador, el diseño de algoritmos y la resolución de problemas, entre otros conceptos que estudiaremos.</p> <p>Al estudiar este tema, los objetivos que se pretenden conseguir son:</p> <ul style="list-style-type: none"><li>· Conocer la estructura básica de un ordenador.</li><li>· Identificar las generalidades sobre el software.</li><li>· Describir distintos lenguajes de programación.</li><li>· Comprender el ciclo de vida de una aplicación informática.</li><li>· Diseñar algoritmos para la resolución de problemas.</li><li>· Estudiar las distintas técnicas descriptivas para representar algoritmos.</li></ul>

SEMANA	CONTENIDO	
		<ul style="list-style-type: none"> <li>· Entender los elementos y la estructura de un algoritmo.</li> </ul>
	Temas	<p>Introducción a la programación y diseño de algoritmos</p> <ul style="list-style-type: none"> <li>· Estructura básica de un ordenador</li> <li>· Software</li> <li>· Lenguajes de programación</li> <li>· Ciclo de vida de una aplicación informática</li> <li>· Diseño de algoritmos y resolución de problemas</li> <li>· Técnicas descriptivas</li> <li>· Elementos y estructura de un algoritmo</li> </ul>
	Actividades	<p>Lectura, estudio, y comprensión de los materiales de la asignatura, incluyendo webinars y la participación obligatoria en los foros, interacción con el profesor y los demás estudiantes. Las actividades interactivas de esta unidad son:</p> <ul style="list-style-type: none"> <li>· Clase de presentación de la asignatura y del tema 1.</li> <li>· Test de tema 1.</li> </ul>
	Lectura adicional y actividades fuera del horario de clase	<p>El profesor puede asignar actividades fuera del aula que figurarán en la calificación final del estudiante (p. ej. Proyectos, casos prácticos, presentaciones, etc.). Estas actividades se comunicarán y se especificarán al estudiante a lo largo de la asignatura.</p> <p>Fuera de clase: 6 horas</p>
<b>Semana 2</b>	Objetivos específicos	<p>Existen muchos lenguajes de programación que pueden ser utilizados para escribir programas, incluso hay de distintos tipos orientados a la resolución de un problema u otro. En este tema, nos centraremos en conocer el lenguaje de programación Java.</p> <p>Al finalizar este tema, el alumno podrá:</p> <ul style="list-style-type: none"> <li>· Conocer el origen y características del lenguaje Java.</li> <li>· Entender las fases de desarrollo de un programa en Java.</li> <li>· Comprender el concepto de un programa escrito en Java.</li> <li>· Aprender los tipos de datos fundamentales utilizados en Java.</li> <li>· Distinguir entre lo que son operadores, expresiones y sentencias para realizar operaciones.</li> <li>· Saber las funciones a utilizar para generar entrada y salida a un programa</li> </ul>
	Temas	<p>Elementos de un programa</p> <ul style="list-style-type: none"> <li>· Origen y características del lenguaje Java</li> <li>· El entorno de desarrollo</li> <li>· Concepto de un programa</li> <li>· Tipos de datos fundamentales</li> </ul>

SEMANA	CONTENIDO	
		<ul style="list-style-type: none"> <li>· Operadores</li> <li>· Expresiones</li> <li>· Sentencias</li> <li>· Entrada y salida de datos</li> </ul>
	Actividades	<p>Lectura, estudio, y comprensión de los materiales de la asignatura, incluyendo webinars y la participación obligatoria en los foros, interacción con el profesor y los demás estudiantes. Las actividades interactivas de esta unidad son:</p> <ul style="list-style-type: none"> <li>· Inicio de la actividad 1.</li> <li>· Test de tema 2.</li> </ul>
	Lectura adicional y actividades fuera del horario de clase	<p>El profesor puede asignar actividades fuera del aula que figurarán en la calificación final del estudiante (p. ej. Proyectos, casos prácticos, presentaciones, etc.). Estas actividades se comunicarán y se especificarán al estudiante a lo largo de la asignatura.</p> <p>Fuera de clase: 6 horas</p>
<b>Semana 3</b>	Objetivos específicos	<p>Las sentencias de control de flujo forman parte de los componentes esenciales de los lenguajes de programación, ya que a través de ellas se tiene control de las acciones y cálculos a ejecutar. Las sentencias admiten la implementación de los algoritmos y determinan el orden en que se ejecutan las demás sentencias en el programa.</p> <p>Al finalizar este tema, el alumno podrá:</p> <ul style="list-style-type: none"> <li>· Conocer las distintas sentencias de control en Java.</li> <li>· Ejecutar las diferentes bifurcaciones según corresponda.</li> <li>· Trabajar con bucle según las condiciones establecidas en el programa.</li> </ul>
	Temas	<p>Sentencias de control</p> <ul style="list-style-type: none"> <li>· Sentencias</li> <li>· Bifurcaciones</li> <li>· Bucles</li> </ul>
	Actividades	<p>Lectura, estudio, y comprensión de los materiales de la asignatura, incluyendo webinars y la participación obligatoria en los foros, interacción con el profesor y los demás estudiantes. Las actividades interactivas de esta unidad son:</p> <ul style="list-style-type: none"> <li>· Test del tema 3</li> <li>· Foro 1</li> <li>· Clase de presentación de la actividad 1 y del tema 2.</li> </ul>
	Lectura adicional y actividades	<p>El profesor puede asignar actividades fuera del aula que figurarán en la calificación final del estudiante (p. ej. Proyectos, casos prácticos, presentaciones, etc.). Estas</p>

SEMANA	CONTENIDO	
	fuera del horario de clase	<p>actividades se comunicarán y se especificarán al estudiante a lo largo de la asignatura.</p> <p>Fuera de clase: 6 horas</p>
<b>Semana 4</b>	Objetivos específicos	<p>En programación modular y programación orientada a objetos es posible dividir en módulos o partes más pequeñas el programa para formar partes independientes una de la otra.</p> <p>Al estudiar este tema, el alumno podrá:</p> <ul style="list-style-type: none"> <li>· Conocer la programación funcional en Java.</li> <li>· Aprender acerca del diseño modular.</li> <li>· Definir el concepto de una función y su utilidad.</li> <li>· Estudiar la definición de una función con un método estático.</li> <li>· Trabajar con el flujo de ejecución de la llamada a una función.</li> <li>· Entender lo que es la devolución de resultados.</li> <li>· Realizar la llamada a una función, es decir, en uso de parámetros.</li> <li>· Saber pasar parámetros por referencia y valor.</li> </ul>
	Temas	<p>Abstracción y modularidad: funciones</p> <ul style="list-style-type: none"> <li>· Programación funcional en Java</li> <li>· Diseño modular</li> <li>· Concepto de función y utilidad</li> <li>· Definición de una función con un método estático</li> <li>· Flujo de ejecución de la llamada a una función</li> </ul>
	Actividades	<p>Lectura, estudio, y comprensión de los materiales de la asignatura, incluyendo webinars y la participación obligatoria en los foros, interacción con el profesor y los demás estudiantes. Las actividades interactivas de esta unidad son:</p> <ul style="list-style-type: none"> <li>· Clase de los temas 3 y 4</li> </ul>
	Lectura adicional y actividades fuera del horario de clase	<p>El profesor puede asignar actividades fuera del aula que figurarán en la calificación final del estudiante (p. ej. Proyectos, casos prácticos, presentaciones, etc.). Estas actividades se comunicarán y se especificarán al estudiante a lo largo de la asignatura.</p> <p>Fuera de clase: 6 horas</p>
<b>Semana 5</b>	Objetivos específicos	<p>En programación modular y programación orientada a objetos es posible dividir en módulos o partes más pequeñas el programa para formar partes independientes una de la otra.</p> <p>Al estudiar este tema, el alumno podrá:</p> <ul style="list-style-type: none"> <li>· Conocer la programación funcional en Java.</li> <li>· Aprender acerca del diseño modular.</li> </ul>

SEMANA	CONTENIDO	
		<ul style="list-style-type: none"> <li>· Definir el concepto de una función y su utilidad.</li> <li>· Estudiar la definición de una función con un método estático.</li> <li>· Trabajar con el flujo de ejecución de la llamada a una función.</li> <li>· Entender lo que es la devolución de resultados.</li> <li>· Realizar la llamada a una función, es decir, en uso de parámetros.</li> </ul> <p>Saber pasar parámetros por referencia y valor.</p>
	Temas	<p>Abstracción y modularidad: funciones (continuación)</p> <ul style="list-style-type: none"> <li>· Devoluciones de resultados</li> <li>· Llamada a una función: parámetros</li> <li>· Paso de parámetros por referencia y por valor</li> </ul>
	Actividades	<p>Lectura, estudio, y comprensión de los materiales de la asignatura, incluyendo webinars y la participación obligatoria en los foros, interacción con el profesor y los demás estudiantes. Las actividades interactivas de esta unidad son:</p> <ul style="list-style-type: none"> <li>· Foro 2.</li> <li>· Test del tema 4.</li> </ul>
	Lectura adicional y actividades fuera del horario de clase	<p>El profesor puede asignar actividades fuera del aula que figurarán en la calificación final del estudiante (p. ej. Proyectos, casos prácticos, presentaciones, etc.). Estas actividades se comunicarán y se especificarán al estudiante a lo largo de la asignatura.</p> <p>Fuera de clase: 6 horas</p>
<b>Semana 6</b>	Objetivos específicos	<p>En este tema estudiaremos lo que es una clase en Java, así como los atributos que puede contener una clase, la manera en la que se crean los objetos y otros conceptos relacionados de gran importancia.</p> <p>Una vez estudiado este tema, el alumno conocerá:</p> <ul style="list-style-type: none"> <li>· El concepto de una clase.</li> <li>· Cómo se crea una clase.</li> <li>· El uso y creación de objetos.</li> <li>· Los atributos de una clase.</li> <li>· El uso de métodos en Java.</li> <li>· La utilidad de los constructores en una clase.</li> </ul>
	Temas	<p>Clases en Java</p> <ul style="list-style-type: none"> <li>· Concepto de una clase</li> <li>· Creación de una clase</li> <li>· Objetos</li> <li>· Creación de objetos</li> </ul>

SEMANA	CONTENIDO	
		<ul style="list-style-type: none"> <li>· Objetos múltiples</li> <li>· Atributos de una clase</li> </ul>
	Actividades	<p>Lectura, estudio, y comprensión de los materiales de la asignatura, incluyendo webinars y la participación obligatoria en los foros, interacción con el profesor y los demás estudiantes. Las actividades interactivas de esta unidad son:</p> <ul style="list-style-type: none"> <li>· Clase del tema 5.</li> </ul>
	Lectura adicional y actividades fuera del horario de clase	<p>El profesor puede asignar actividades fuera del aula que figurarán en la calificación final del estudiante (p. ej. Proyectos, casos prácticos, presentaciones, etc.). Estas actividades se comunicarán y se especificarán al estudiante a lo largo de la asignatura.</p> <p>Fuera de clase: 6 horas</p>
<b>Semana 7</b>	Objetivos específicos	<p>En este tema estudiaremos lo que es una clase en Java, así como los atributos que puede contener una clase, la manera en la que se crean los objetos y otros conceptos relacionados de gran importancia.</p> <p>Una vez estudiado este tema, el alumno conocerá:</p> <ul style="list-style-type: none"> <li>· El concepto de una clase.</li> <li>· Cómo se crea una clase.</li> <li>· El uso y creación de objetos.</li> <li>· Los atributos de una clase.</li> <li>· El uso de métodos en Java.</li> <li>· La utilidad de los constructores en una clase.</li> </ul>
	Temas	<p>Clases en Java</p> <ul style="list-style-type: none"> <li>· Atributos múltiples</li> <li>· Concepto de método y utilidad</li> <li>· Métodos estáticos y no estáticos</li> <li>· Constructores</li> </ul>
	Actividades	<p>Lectura, estudio, y comprensión de los materiales de la asignatura, incluyendo webinars y la participación obligatoria en los foros, interacción con el profesor y los demás estudiantes. Las actividades interactivas de esta unidad son:</p> <ul style="list-style-type: none"> <li>· Entrega de la actividad 1.</li> <li>· Fin foro 1.</li> <li>· Test del tema 5.</li> </ul>
	Lectura adicional y actividades fuera del horario de clase	<p>El profesor puede asignar actividades fuera del aula que figurarán en la calificación final del estudiante (p. ej. Proyectos, casos prácticos, presentaciones, etc.). Estas actividades se comunicarán y se especificarán al estudiante a lo largo de la asignatura.</p>

SEMANA	CONTENIDO	
		Fuera de clase: 6 horas
<b>Semana 8:</b>	Objetivos específicos	<p>Entender cómo se resuelven las ligaduras en función de la relación establecida entre clases.</p> <p>Un conjunto de datos que tienen el mismo tipo es una estructura de datos homogénea. Si la estructura de datos reside en la memoria principal, es una estructura de datos interna, y si reside en un soporte externo, como puede ser un disco duro, es una estructura de datos externa.</p> <p>Asimismo, una estructura estática tiene un número fijo de elementos, mientras que una estructura dinámica tiene un número variable de elementos durante la ejecución del programa.</p>
	Temas	<p>Estructuras de datos en Java</p> <ul style="list-style-type: none"> <li>· Arrays</li> <li>· Matrices</li> <li>· Cadenas Funciones de E/S para cadenas</li> <li>· Clases ArrayList</li> </ul>
	Actividades	<p>Lectura, estudio, y comprensión de los materiales de la asignatura, incluyendo webinars y la participación obligatoria en los foros, interacción con el profesor y los demás estudiantes. Las actividades interactivas de esta unidad son:</p> <ul style="list-style-type: none"> <li>· Clase de resolución de la actividad 1 y presentación del tema 6</li> <li>· Test del tema 6.</li> </ul>
	Lectura adicional y actividades fuera del horario de clase	<p>El profesor puede asignar actividades fuera del aula que figurarán en la calificación final del estudiante (p. ej. Proyectos, casos prácticos, presentaciones, etc.). Estas actividades se comunicarán y se especificarán al estudiante a lo largo de la asignatura.</p> <p>Fuera de clase: 6 horas</p>
<b>Semana 9:</b>	Objetivos específicos	<p>Aquí vamos a estudiar las principales relaciones entre clases, profundizando las implicaciones que esas relaciones tienen cuando programamos. Asimismo, explicaremos cómo utilizar estos conceptos en la parte de diseño y cómo reflejarlos en un diagrama de clases.</p> <p>Por tanto, en este tema pretendemos:</p> <ul style="list-style-type: none"> <li>· Identificar las relaciones que pueden darse entre clases.</li> <li>· Comprender las implicancias de las relaciones que se dan entre clases tanto en el diseño como en la programación.</li> <li>· Entender cómo se resuelven las ligaduras en función de la relación establecida entre clases.</li> </ul>

SEMANA	CONTENIDO	
	Temas	<p>Relaciones entre clases</p> <ul style="list-style-type: none"> <li>· Herencia</li> <li>· Polimorfismo</li> <li>· This y super</li> <li>· Clases abstractas e interfaces</li> </ul>
	Actividades	<p>Lectura, estudio, y comprensión de los materiales de la asignatura, incluyendo webinars y la participación obligatoria en los foros, interacción con el profesor y los demás estudiantes. Las actividades interactivas de esta unidad son:</p> <ul style="list-style-type: none"> <li>· Clase de presentación de la actividad 2 y del tema 7.</li> <li>· Test del tema 7</li> </ul>
	Lectura adicional y actividades fuera del horario de clase	<p>El profesor puede asignar actividades fuera del aula que figurarán en la calificación final del estudiante (p. ej. Proyectos, casos prácticos, presentaciones, etc.). Estas actividades se comunicarán y se especificarán al estudiante a lo largo de la asignatura.</p> <p>Fuera de clase: 6 horas</p>
<b>Semana 10:</b>	Objetivos específicos	<p>Un fichero es una colección de registros relacionados lógicamente, con una longitud determinada que reside en un almacenamiento externo asociado a una computadora.</p>
	Temas	<p>Ficheros</p> <ul style="list-style-type: none"> <li>· Conceptos básicos</li> <li>· Operaciones con ficheros</li> <li>· Tipos de ficheros</li> <li>· Organización de los ficheros</li> <li>· Introducción a los ficheros en Java</li> <li>· Manejo de ficheros</li> </ul>
	Actividades	<p>Lectura, estudio, y comprensión de los materiales de la asignatura, incluyendo webinars y la participación obligatoria en los foros, interacción con el profesor y los demás estudiantes. Las actividades interactivas de esta unidad son:</p> <ul style="list-style-type: none"> <li>· Test del tema 8.</li> <li>· Clase del tema 8.</li> </ul>
	Lectura adicional y actividades fuera del horario de clase	<p>El profesor puede asignar actividades fuera del aula que figurarán en la calificación final del estudiante (p. ej. Proyectos, casos prácticos, presentaciones, etc.). Estas actividades se comunicarán y se especificarán al estudiante a lo largo de la asignatura.</p> <p>Fuera de clase: 6 horas</p>

SEMANA	CONTENIDO	
<b>Semana 11:</b>	Objetivos específicos	<p>Al principio, los programadores piensan en la recursividad como una técnica confusa o difícil, pero lo cierto es que es la forma más sencilla de resolver algunos problemas, como los relacionados con estructuras jerárquicas. Por eso, en este tema vamos a hablar de las bases de la recursividad y vamos a ponerla en práctica usando el lenguaje Java.</p> <p>Al final de este tema, serás capaz de:</p> <ul style="list-style-type: none"> <li>· Entender qué es la recursividad.</li> <li>· Identificar diferentes tipos de recursividad.</li> <li>· Entender cómo funciona la recursividad, poniéndola en práctica en algunos problemas de uso típico.</li> </ul>
	Temas	<p>Recursividad</p> <ul style="list-style-type: none"> <li>· Definición de recursividad</li> <li>· Tipos de recursión</li> <li>· Ventajas e inconvenientes</li> <li>· Consideraciones</li> <li>· Conversión recursivo-iterativa</li> <li>· La pila de recursión</li> </ul>
	Actividades	<p>Lectura, estudio, y comprensión de los materiales de la asignatura, incluyendo webinars y la participación obligatoria en los foros, interacción con el profesor y los demás estudiantes. Las actividades interactivas de esta unidad son:</p> <ul style="list-style-type: none"> <li>· Foro 3.</li> <li>· Test del tema 9.</li> </ul>
	Lectura adicional y actividades fuera del horario de clase	<p>El profesor puede asignar actividades fuera del aula que figurarán en la calificación final del estudiante (p. ej. Proyectos, casos prácticos, presentaciones, etc.). Estas actividades se comunicarán y se especificarán al estudiante a lo largo de la asignatura.</p> <p>Fuera de clase: 6 horas</p>
<b>Semana 12:</b>	Objetivos específicos	<p>Cuando desarrollamos, sobre todo en este contexto didáctico, hacemos pruebas para asegurarnos de que nuestro código funciona. Sin embargo, esto que hacemos por inercia es algo que está estudiado, medido y documentado. Es un proceso importante del desarrollo de software y ocupa una parte importante del tiempo del proyecto. Por eso, al finalizar este tema, habrás satisfecho los siguientes objetivos:</p> <ul style="list-style-type: none"> <li>· Entender la relevancia de las pruebas del software.</li> <li>· Distinguir los distintos tipos de pruebas.</li> <li>· Construir distintos casos de prueba para diferentes situaciones.</li> </ul>

SEMANA	CONTENIDO	
		<ul style="list-style-type: none"> <li>Comprender la necesidad de pruebas adicionales para POO.</li> </ul>
	Temas	Prueba y documentación <ul style="list-style-type: none"> <li>Pruebas de programas</li> <li>Prueba de la caja blanca</li> <li>Prueba de la caja negra</li> <li>Herramientas para realizar las pruebas</li> <li>Documentación de programas</li> <li>Pruebas específicas para POO</li> </ul>
	Actividades	Lectura, estudio, y comprensión de los materiales de la asignatura, incluyendo webinars y la participación obligatoria en los foros, interacción con el profesor y los demás estudiantes. Las actividades interactivas de esta unidad son: <ul style="list-style-type: none"> <li>Clase de los temas 9 y 10.</li> <li>Test del tema 10.</li> </ul>
	Lectura adicional y actividades fuera del horario de clase	El profesor puede asignar actividades fuera del aula que figurarán en la calificación final del estudiante (p. ej. Proyectos, casos prácticos, presentaciones, etc.). Estas actividades se comunicarán y se especificarán al estudiante a lo largo de la asignatura. Fuera de clase: 6 horas
<b>Semana 13:</b>	Objetivos específicos	
	Temas	
	Actividades	Lectura, estudio, y comprensión de los materiales de la asignatura, incluyendo webinars y la participación obligatoria en los foros, interacción con el profesor y los demás estudiantes. Las actividades interactivas de esta unidad son: <ul style="list-style-type: none"> <li>Entrega de la actividad 2.</li> <li>Clase de repaso de contenidos y/o resolución de ejercicios.</li> </ul>
	Lectura adicional y actividades fuera del horario de clase	El profesor puede asignar actividades fuera del aula que figurarán en la calificación final del estudiante (p. ej. Proyectos, casos prácticos, presentaciones, etc.). Estas actividades se comunicarán y se especificarán al estudiante a lo largo de la asignatura. Fuera de clase: 6 horas
<b>Semana 14:</b>	Objetivos específicos	
	Temas	

SEMANA	CONTENIDO	
	Actividades	Lectura, estudio, y comprensión de los materiales de la asignatura, incluyendo webinars y la participación obligatoria en los foros, interacción con el profesor y los demás estudiantes. Las actividades interactivas de esta unidad son: <ul style="list-style-type: none"> <li>· Clase de resolución de la actividad 2 y repaso</li> </ul>
	Lectura adicional y actividades fuera del horario de clase	El profesor puede asignar actividades fuera del aula que figurarán en la calificación final del estudiante (p. ej. Proyectos, casos prácticos, presentaciones, etc.). Estas actividades se comunicarán y se especificarán al estudiante a lo largo de la asignatura. Fuera de clase: 6 horas
<b>Semana 15:</b>	Objetivos específicos	
	Temas	
	Actividades	Lectura, estudio, y comprensión de los materiales de la asignatura, incluyendo webinars y la participación obligatoria en los foros, interacción con el profesor y los demás estudiantes. Las actividades interactivas de esta unidad son: <ul style="list-style-type: none"> <li>· Examen final (respuesta abierta).</li> </ul>
	Lectura adicional y actividades fuera del horario de clase	El profesor puede asignar actividades fuera del aula que figurarán en la calificación final del estudiante (p. ej. Proyectos, casos prácticos, presentaciones, etc.). Estas actividades se comunicarán y se especificarán al estudiante a lo largo de la asignatura. Fuera de clase: 6 horas

## **Metodología**

A la hora de desarrollar estrategias metodológicas, es conveniente comentarlas entre profesores y estudiantes en un entorno abierto y de apoyo para asegurarse de que los estudiantes tomen la responsabilidad por su implementación y por lograr los objetivos de la asignatura.

Las siguientes estrategias pueden utilizarse en esta asignatura:

- ▶ Un repaso de las preguntas al final de cada capítulo.
- ▶ Comprobación de comprensión de lectura.
- ▶ Análisis de lecturas asignadas.
- ▶ Discusiones en grupo.
- ▶ Discusiones individuales y en grupo.
- ▶ Preparación de reportes.
- ▶ Creación de un plan de enseñanza.
- ▶ Llevar a cabo una microclase.

## **Materiales recomendados**

- ▶ Hu, Y. (2020). Fácil Aprendizaje. Estructuras de Datos y Algoritmos Java. Michigan: Independently published.
- ▶ Jiménez Marín, A. y Pérez Montes, F. M. (2016). Aprende a Programar con Java. (2.<sup>a</sup> ed.). Madrid: Ediciones Paraninfo, S.A.
- ▶ Joyanes Aguilar, L. (2020). Fundamentos de Programación. Algoritmos, Estructura de Datos y Objetos (5.<sup>a</sup> ed.). Madrid: McGraw-Hill.
- ▶ Sedgewick, R. y Wayne, K. (2017). Introduction to Programming in Java. (2.<sup>a</sup> ed.). Londres: Sams Div Of Pearson.

## Crterios y mtodos de evaluaci3n de estudiantes

Letra	Valor num3rico	GPA
A	97 – 100%	4,0
A-	90 - 96%	3,7
B+	87 – 89%	3,3
B	80 – 86%	3,0
B-	78 – 79%	2,7
C+	75 – 77%	2,3
C	70 – 74%	2,0
C-	67 – 69%	1,7
D+	63 – 66%	1,3
D	57 – 62%	1,0
F	< 57%	0,0
I	-	Incomplete*
TR	-	Transfer Credit**
W	-	Withdrawal**
WP	-	Withdraw Passing**
WF	0	Withdraw Failing

\* Nota no se calcula como parte del CGPA del estudiante, pero las horas de cr3dito se incluyen en el total de cr3ditos intentados.

\*\* Nota no se calcula como parte del CGPA del estudiante, y las horas de cr3dito no se incluyen en el total de cr3ditos intentados.

### CALIFICACIONES APROBATORIAS

Para programas de ASSOCIATE Y BACHELOR'S, la nota aprobatoria es de C (2,0) o m3s.

Para programas de M3STER, la nota aprobatoria es de B (3,0) o m3s.

### Distribuci3n de calificaciones

Evaluaci3n de la asignatura	Peso
Foros	15%
Evaluaci3n de actividades interactivas (a trav3s de los foros)	35%
Examen de cada tema	20%
Examen Final (respuesta abierta)	30%
<b>Total</b>	<b>100%</b>

Última revisi3n del syllabus: MAYO 2022