

## Programa de asignatura por competencias de educación superior

### Sección I. Identificación del Curso

Tabla 1. Identificación de la Planificación del Curso.

<b>Actualización:</b>	Marzo 22, 2022				
<b>Carrera:</b>	Ingeniería en Tecnologías de Software	<b>Asignatura:</b>	Programación Orientada a Objetos		
<b>Academia:</b>	Informática y Computación Virtual /	<b>Clave:</b>	19SICTS0406		
<b>Módulo formativo:</b>	Informática y Computación	<b>Seriación:</b>	19SICTS0507 - Arquitectura de Software		
<b>Tipo de curso:</b>	Modalidad mixta	<b>Prerrequisito:</b>	19SICTS0303 - Programación Estructurada		
<b>Semestre:</b>	Cuarto	<b>Créditos:</b>	5.63	<b>Horas semestre:</b>	90 horas
<b>Teoría:</b>	1 hora	<b>Práctica:</b>	1 hora	<b>Trabajo indpt.:</b>	3 horas
				<b>Total x semana:</b>	5 horas

## Sección II. Objetivos educacionales

Tabla 2. Objetivos educacionales

Objetivos educacionales		Criterios de desempeño	Indicadores
OE1	Solucionará problemas con sólidas bases científicas y fundamentos tecnológicos que le permitirán comprender, analizar, diseñar, organizar, producir, operar y dar soluciones prácticas a problemas relacionados con las áreas de Organización de Sistemas Computacionales e Ingeniería en Software para el sector productivo y social, promoviendo los principios de ética, responsabilidad y trabajo colaborativo.	CD1. El egresado implementará las diferentes etapas del ciclo de vida del software contemplando la protección de datos y prevención dedesastres, salvaguardando con ética la seguridad de la información.	50 % Egresados trabajarán en cualquier proceso del desarrollo de software o áreas afines a los sistemas computacionales, promoviendo los principios de ética, responsabilidad y trabajo colaborativo.
OE2	Aportará soluciones innovadoras y sustentables en el área de la electrónica en el que establezca el análisis, diseño, implementación, selección de componentes de hardware de uso específico, el software asociado y su conectividad a través de redes de comunicación para el sector productivo y social.	CD2. El egresado implementará las diferentes técnicas de análisis y diseño de circuitos electrónicos que den una solución innovadorasustentable a problemas con el hardware.	20% Egresados trabajarán en cualquier proceso de creación y aplicación de hardware o áreas afines en el sector productivo y social.
OE3	Implementará soluciones innovadoras y sustentables con tecnologías de información que sean acordes a las necesidades, a las tecnologías disponibles y emergentes, para lograr un aprovechamiento óptimo de los recursos humanos y financieros en el sector productivo y social.	CD3. El egresado implementará las diferentes tecnologías emergentes en equipos multidisciplinarios que den una solución innovadora ysustentable a las necesidades que se presenten en el ámbito productivo y social.	20 % Egresados trabajarán en la aplicación de Tecnologías de la información o áreas afines en el sector productivo o social.



Atributos de egreso de plan de estudios		Criterios de desempeño	Componentes
AE2	Gestionar diferentes productos y aplicaciones de software que cumplan las necesidades de los sectores públicos y privados.	<p>CD1. Comprenderá el entorno de desarrollo para la programación orientada a objetos que se utilizará en el diseño de programas para una solución efectiva en la resolución de problemas específicos.</p> <p>CD2. Representará los diferentes diagramas UML que dan solución a un problema específico teniendo una panorámica de la respuesta a programar.</p> <p>CD3. Comprenderá las clases y métodos para su aplicación en la programación orientada a objetos en el diseño de programas básicos para dar una solución efectiva en la resolución de problemas específicos.</p> <p>CD4. Diseñará programas con métodos y su aplicación en la programación orientada a objetos en el diseño de programas específicos para la solución de problemas básicos.</p> <p>CD5. Diseñará programas con herencia y polimorfismo, así como su aplicación en la programación orientada a objetos en el diseño de programas específicos para la solución de problemas con requerimientos con complejidad mayor.</p> <p>CD6. Diseñará programas con arreglos y su aplicación en la programación orientada a objetos en el diseño de programas específicos para la solución de problemas con requerimientos de almacenamiento de datos.</p> <p>CD7. Diseñará programas con archivos y su aplicación en la programación orientada a objetos en el diseño de programas con requerimientos de almacenamiento permanente.</p>	<p>1. Entorno de desarrollo de la POO. Paradigma de Programación orientado a objetos. Lenguajes de POO. Estructura de un proyecto.</p> <p>1.4 Proceso de desarrollo de aplicaciones. 1.4.1 Compilación. 1.4.2 Archivos que crea el entorno de desarrollo. 1.4.3 Depuración.</p> <p>2. Diagramas UML. 2.1. Conceptos básicos de POO. 2.2 Diagramas de Casos de uso. 2.3. Diagramas de Clases. 2.4 Diagramas de Actividades. 2.5. Diagramas de secuencia.</p> <p>3 Clases y objetos. 3.1 Definición de una clase. 3.2 Declaración de clases. 3.3 Miembros de una clase. 3.4 Ámbito referente a una clase. 3.5 Especificadores de acceso. 3.6 Creación de objetos. 3.7 Clases predefinidas. 3.8 Definición, creación y reutilización de paquetes/librerías. 3.9 Manejo de excepciones.</p> <p>4 Métodos.</p>



Continuación: Tabla 2. Objetivos educacionales (continuación)

No.	Atributos de egreso de plan de estudios	Criterios de desempeño	Componentes
			4.1 Definición de un método. 4.2 Estructura de un método. 4.3 Valor de retorno. 4.4 Declaración de un método. 4.4.1 De clase. 4.4.2 De Instancia. 4.5 Ámbito y tiempo de vida de variables. 4.6 Argumentos y paso de parámetros. 4.7 Puntero this. 4.8 Sobrecarga de métodos. 4.9 Constructores y destructores. 5 Herencia y polimorfismo. 5.1 Concepto de herencia y polimorfismo. 5.2 Definición de una clase base. 5.3 Definición de una clase derivada. 5.4 Clases abstractas. 5.5 Interfaces. 5.5.1 Definición. 5.5.2 Implementación. 5.5.3 Variables polimórficas. 5. 6 Reutilización de la definición de paquetes/librerías. 6 Arreglos. 6.1 Unidimensional. 6.2 Multidimensional. 7 Archivos. 7.1 Definición de Archivos de texto y archivos binarios. 7.2 Operaciones básicas en archivos texto y binario. 7.3 Manejo de excepciones en archivos. 8. Aplicaciones Gráficas. 81. Principales componentes y eventos.



Continuación: Tabla 2. Objetivos educacionales (continuación)

No.	Atributos de egreso de plan de estudios	Criterios de desempeño	Componentes
			8.2 Menús. 8.3 Gráficos e imágenes. 8.4 Diseño e implementación de aplicaciones OO.



### Sección III. Atributos de la asignatura

Tabla 3. Atributos de la asignatura

Problema a resolver		
Iniciar a los estudiantes en la programación orientada a objetos, partiendo del entorno de desarrollo para el diseño del programa que represente la solución planteada.		
Atributos (competencia específica) de la asignatura		
Desarrollar programas que permitan almacenar, disponer y procesar información, utilizando el paradigma de programación orientada a objetos que permitirán la construcción de programas con el uso de las diferentes estructuras que se manejan.		
Aportación a la competencia específica		Aportación a las competencias transversales
Saber	Saber hacer	Saber Ser
<ul style="list-style-type: none"> <li>- Conocer las estructuras del lenguaje orientada a objetos.</li> <li>- Analizar las bases de la programación orientada a objetos.</li> <li>- Identificar las diferentes partes del entorno de desarrollo.</li> <li>- Analizar y sintetizar soluciones a problemas planteados.</li> </ul>	<ul style="list-style-type: none"> <li>- Resolver problemas con las diferentes estructuras del lenguaje orientada a objetos.</li> <li>- Aplicar los conocimientos en la práctica en el desarrollo de programas orientados a objetos.</li> <li>- Plantear y resolver problemas específicos acordes a las bases de la programación orientada a objetos.</li> </ul>	<ul style="list-style-type: none"> <li>- Aporta puntos de vista con apertura a aprender de los otros y considera los de otras personas de manera reflexiva y respetuosa.</li> <li>- Participa activamente en la construcción de su aprendizaje y en la resolución de problemas, colaborando de manera productiva en espacios y equipos de trabajo.</li> <li>- Entrega en tiempo y forma sus actividades de aprendizaje siguiendo las indicaciones y considerando los criterios de evaluación.</li> <li>- Utiliza la tecnología para apoyar su aprendizaje y para el desarrollo de habilidades metacognitivas, el aprendizaje autónomo y el longlife learning.</li> </ul>
Producto integrador de la asignatura, considerando los avances por unidad		
Proyecto integrador, a partir de la creación de un programa, dando solución a una necesidad real en una organización, incorporando las competencias desarrolladas en cada una de las unidades de aprendizaje.		

## Sección IV. Desglose específico por cada unidad formativa

Tabla 4.1. Desglose específico de la unidad "Entorno de desarrollo de la POO."

<b>Número y nombre de la unidad:</b> 1. Entorno de desarrollo de la POO.							
<b>Tiempo y porcentaje para esta unidad:</b>		Teoría:	4 horas	Práctica:	4 horas	Porcentaje del programa:	8.89%
<b>Aprendizajes esperados:</b> Conocer el entorno de desarrollo para el diseño de un programa orientado a objetos.							
Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad (Evidencia de aprendizaje de la unidad)			
1 Entorno de desarrollo de la POO. 1.1 Paradigma de Programación orientado a objetos. 1.2 Lenguajes de POO. 1.3. Estructura de un proyecto. 1.4 Proceso de desarrollo de aplicaciones. 1.4.1 Compilación. 1.4.2 Archivos que crea el entorno de desarrollo. 1.4.3 Depuración.	Saber: - Identificar el entorno de desarrollo para el diseño de un programa orientado a objetos y sus bases.  Saber hacer: - Demostrar el manejo del entorno de desarrollo en el diseño de programas.  Ser: - Aporta puntos de vista con apertura a aprender de los otros y considera los de otras personas de manera reflexiva y respetuosa. - Participa activamente en la construcción de su aprendizaje y en la resolución de	- Exposición por parte del profesor de material teórico. - Complementar información con material audiovisual. - Revisar la Introducción a la programación orientada a objetos mediante ejercicios prácticos y básicos. - Realizar ejercicios programados para las actividades del desarrollo de una aplicación. - Realizar tareas de ejercicios de programas.	Evaluación Diagnóstica: - Rescatar conocimientos previos. Instrumento: cuestionario.  Evaluación Formativa: - Realización de ejercicios prácticos. Instrumento: Rúbrica.  Evaluación sumativa: - Elaborar el planteamiento de un problema laboral o cotidiano. Instrumento: Rúbrica o lista de cotejo.	Demostración del uso del entorno de desarrollo para la creación, compilación y ejecución de un programa orientado a objetos.			



Continuación: Tabla 4.1. Desglose específico de la unidad "Entorno de desarrollo de la POO."

Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad
	<p>problemas, colaborando de manera productiva en espacios y equipos de trabajo.</p> <ul style="list-style-type: none"> <li>- Entrega en tiempo y forma sus actividades de aprendizaje siguiendo las indicaciones y considerando los criterios de evaluación.</li> <li>- Utiliza la tecnología para apoyar su aprendizaje y para el desarrollo de habilidades metacognitivas, el aprendizaje autónomo y el longlife learning.</li> </ul>			
<b>Bibliografía</b>				
<p>Moreno, J.C. (2014), Programación Orientada a Objetos. España: Rama</p> <p>Sznajdleder, P. (2017). Programación orientada a objetos y estructura de datos a fondo Implementación de algoritmos en Java, Alfa omega.</p> <p>Deitel, P. J. ; Harvey, M. D. (2016). Cómo programar en java. México: Pearson educación.</p>				



## Sección IV. Desglose específico por cada unidad formativa

Tabla 4.2. Desglose específico de la unidad "Diagramas UML."

Número y nombre de la unidad: 2. Diagramas UML.							
Tiempo y porcentaje para esta unidad:		Teoría:	4 horas	Práctica:	6 horas	Porcentaje del programa:	11.11%
Aprendizajes esperados:		Diseñar los diagramas UML para representar la solución de un problema planteado.					
Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad (Evidencia de aprendizaje de la unidad)			
2. Diagramas UML. 2.1. Conceptos básicos de POO. 2.2 Diagramas de Casos de uso. 2.3. Diagramas de Clases. 2.4 Diagramas de Actividades. 2.5. Diagramas de secuencia.	Saber: - Identificar y analizar los diagramas UML para representar la solución a un problema.  Saber hacer: - Aplicar los conceptos y elementos de los diagramas UML para resolver problemas de ingeniería.  Ser: - Aporta puntos de vista con apertura a aprender de los otros y considera los de otras personas de manera reflexiva y respetuosa.	-Exposición por parte del docente con apoyo de las presentaciones con los temas descritos. - Complementar información con material audiovisual. -Resolver el problemario de ejercicios para diseñar diagramas -Realizar ejercicios programados para las actividades de cada diagrama. - Realizar tareas de ejercicios de diagramas.	- Evaluación Formativa: Realización de ejercicios prácticos. Instrumento: Rúbrica.  - Evaluación sumativa: Elaborar el planteamiento de un problema laboral o cotidiano representado en diagramas UML. Instrumento: Rúbrica o lista de cotejo.	Planteamiento de un problema laboral o cotidiano en el que se pueda aplicar los diferentes diagramas UML que representen la solución.			



Continuación: Tabla 4.2. Desglose específico de la unidad "Diagramas UML."

Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad
	<ul style="list-style-type: none"> <li>- Participa activamente en la construcción de su aprendizaje y en la resolución de problemas, colaborando de manera productiva en espacios y equipos de trabajo.</li> <li>- Entrega en tiempo y forma sus actividades de aprendizaje siguiendo las indicaciones y considerando los criterios de evaluación.</li> <li>- Utiliza la tecnología para apoyar su aprendizaje y para el desarrollo de habilidades metacognitivas, el aprendizaje autónomo y el longlife learning.</li> </ul>			
<b>Bibliografía</b>				
<p>Moreno, J.C. (2014), Programación Orientada a Objetos. España: Rama</p> <p>Sznajdleder, P. (2017). Programación orientada a objetos y estructura de datos a fondo Implementación de algoritmos en Java, Alfa omega.</p> <p>Deitel, P. J. ; Harvey, M. D. (2016). Cómo programar en java. México: Pearson educación.</p>				

## Sección IV. Desglose específico por cada unidad formativa

Tabla 4.3. Desglose específico de la unidad "Clases y objetos."

<b>Número y nombre de la unidad:</b> 3. Clases y objetos.				
<b>Tiempo y porcentaje para esta unidad:</b>		Teoría: 6 horas	Práctica: 4 horas	Porcentaje del programa: 11.11%
<b>Aprendizajes esperados:</b>		Comprender las clases y métodos para su aplicación en la programación orientada a objetos en el diseño de programas básicos para dar una solución efectiva en la resolución de problemas específicos.		
Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad (Evidencia de aprendizaje de la unidad)
3 Clases y objetos. 3.1 Definición de una clase. 3.2 Declaración de clases. 3.3 Miembros de una clase. 3.4 Ámbito referente a una clase. 3.5 Especificadores de acceso. 3.6 Creación de objetos. 3.7 Clases predefinidas. 3.8 Definición, creación y reutilización de paquetes/librerías. 3.9 Manejo de excepciones.	<b>Saber:</b> - Analizar los conceptos de clases y objetos.  <b>Saber hacer:</b> - Aplicar las clases y métodos en el diseño de programas básicos  <b>Ser:</b> - Aporta puntos de vista con apertura a aprender de los otros y considera los de otras personas de manera reflexiva y respetuosa. - Participa activamente en la construcción de su aprendizaje y en la resolución de problemas, colaborando de manera	-Exposición por parte del docente con apoyo de las presentaciones con los temas descritos. - Complementar información con material audiovisual. -Resolver el problemario de ejercicios para programar. -Realizar ejercicios programados para las actividades clases y objetos.	<b>- Evaluación Formativa:</b> Realización de ejercicios prácticos. Instrumento: Rúbrica.  <b>- Evaluación sumativa:</b> Elaborar el planteamiento de un problema laboral o cotidiano. Instrumento: Rúbrica o lista de cotejo.	Planteamiento de un problema laboral o cotidiano en el que se pueda aplicar los clases y objetos que represente la solución en un programa.



Continuación: Tabla 4.3. Desglose específico de la unidad "Clases y objetos."

Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad
	<p>productiva en espacios y equipos de trabajo.</p> <ul style="list-style-type: none"> <li>- Entrega en tiempo y forma sus actividades de aprendizaje siguiendo las indicaciones y</li> </ul> <p>considerando los criterios de evaluación.</p> <ul style="list-style-type: none"> <li>- Utiliza la tecnología para apoyar su aprendizaje y para el desarrollo de habilidades metacognitivas, el aprendizaje autónomo y el longlife learning.</li> </ul>			
<b>Bibliografía</b>				
<p>Moreno, J.C. (2014), Programación Orientada a Objetos. España: Rama</p> <p>Sznajdleder, P. (2017). Programación orientada a objetos y estructura de datos a fondo Implementación de algoritmos en Java, Alfa omega.</p> <p>Deitel, P. J. ; Harvey, M. D. (2016). Cómo programar en java. México: Pearson educación.</p>				

## Sección IV. Desglose específico por cada unidad formativa

Tabla 4.4. Desglose específico de la unidad "Implementación de Métodos de instancia y clase."

<b>Número y nombre de la unidad:</b> 4. Implementación de Métodos de instancia y clase.							
<b>Tiempo y porcentaje para esta unidad:</b>		Teoría:	6 horas	Práctica:	4 horas	Porcentaje del programa:	11.11%
<b>Aprendizajes esperados:</b>		Diseñar programas con métodos y su aplicación en la programación orientada a objetos en el diseño de programas específicos para la solución de problemas básicos.					
Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad (Evidencia de aprendizaje de la unidad)			
4 Métodos. 4.1 Definición de un método. 4.2 Estructura de un método. 4.3 Valor de retorno. 4.4 Declaración de un método. 4.4.1 De clase. 4.4.2 De Instancia. 4.5 Ámbito y tiempo de vida de variables. 4.6 Argumentos y paso de parámetros. 4.7 Puntero this. 4.8 Sobrecarga de métodos. 4.9 Constructores y destructores.	Saber: - Analizar los conceptos de métodos, sus tipos e implementación en la POO.  Saber hacer: - Aplicar los métodos de instancia y clase en el diseño de programas básicos.  Ser: - Aporta puntos de vista con apertura a aprender de los otros y considera los de otras personas de manera reflexiva y respetuosa. - Participa activamente en la construcción de su aprendizaje y en la resolución de	-Exposición por parte del docente con apoyo de las presentaciones con los temas descritos.  - Complementar información con material audiovisual. -Resolver el problemario de ejercicios para programar. -Realizar ejercicios programados para las actividades usando métodos de instancia y clase.	- Evaluación Formativa: Realización de ejercicios prácticos. Instrumento: Rúbrica.  - Evaluación sumativa: Elaborar el planteamiento de un problema laboral o cotidiano. Instrumento: Rúbrica o lista de cotejo.	Planteamiento de un problema laboral o cotidiano en el que se pueda aplicar los métodos que represente la solución en un programa.			





Continuación: Tabla 4.4. Desglose específico de la unidad "Implementación de Métodos de instancia y clase."

Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad
	<p>problemas, colaborando de manera productiva en espacios y equipos de trabajo.</p> <ul style="list-style-type: none"> <li>- Entrega en tiempo y forma sus actividades de aprendizaje siguiendo las indicaciones y considerando los criterios de evaluación.</li> <li>- Utiliza la tecnología para apoyar su aprendizaje y para el desarrollo de habilidades metacognitivas, el aprendizaje autónomo y el longlife learning.</li> </ul>			
<b>Bibliografía</b>				
<p>Moreno, J.C. (2014), Programación Orientada a Objetos. España: Rama</p> <p>Sznajdleder, P. (2017). Programación orientada a objetos y estructura de datos a fondo Implementación de algoritmos en Java, Alfa omega.</p> <p>Deitel, P. J. ; Harvey, M. D. (2016). Cómo programar en java. México: Pearson educación.</p>				

## Sección IV. Desglose específico por cada unidad formativa

Tabla 4.5. Desglose específico de la unidad "Herencia y polimorfismo."

<b>Número y nombre de la unidad:</b> 5. Herencia y polimorfismo.				
<b>Tiempo y porcentaje para esta unidad:</b>		Teoría: 6 horas	Práctica: 4 horas	Porcentaje del programa: 11.11%
<b>Aprendizajes esperados:</b>		Diseñar programas con herencia y polimorfismo, así como su aplicación en la programación orientada a objetos en el diseño de programas específicos para la solución de problemas con requerimientos con complejidad mayor.		
Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad (Evidencia de aprendizaje de la unidad)
5 Herencia y polimorfismo. 5.1 Concepto de herencia y polimorfismo. 5.2 Definición de una clase base. 5.3 Definición de una clase derivada. 5.4 Clases abstractas. 5.5 Interfaces. 5.5.1 Definición. 5.5.2 Implementación. 5.5.3 Variables polimórficas. 5.6 Reutilización de la definición de paquetes/librerías.	<b>Saber:</b> - Analizar los conceptos de herencia y polimorfismo.  <b>Saber hacer:</b> - Aplicar la herencia y polimorfismo en el diseño de programas con mayor complejidad  <b>Ser:</b> - Aporta puntos de vista con apertura a aprender de los otros y considera los de otras personas de manera reflexiva y respetuosa. - Participa activamente en la construcción de su aprendizaje y en la resolución de	-Exposición por parte del docente con apoyo de las presentaciones con los temas descritos. - Complementar información con material audiovisual. -Resolver el problemario de ejercicios para programar. -Realizar ejercicios programados para las actividades de herencia. -Realizar tareas de ejercicios de polimorfismo.	- Evaluación Formativa: Realización de ejercicios prácticos. Instrumento: Rúbrica.  - Evaluación sumativa: Elaborar el planteamiento de un problema laboral o cotidiano. Instrumento: Rúbrica o lista de cotejo.	Planteamiento de un problema laboral o cotidiano en el que se pueda aplicar la herencia y polimorfismo que represente la solución en un programa.



Continuación: Tabla 4.5. Desglose específico de la unidad "Herencia y polimorfismo."

Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad
	<p>problemas, colaborando de manera productiva en espacios y equipos de trabajo.</p> <ul style="list-style-type: none"> <li>- Entrega en tiempo y forma sus actividades de aprendizaje siguiendo las indicaciones y considerando los criterios de evaluación.</li> <li>- Utiliza la tecnología para apoyar su aprendizaje y para el desarrollo de habilidades metacognitivas, el aprendizaje autónomo y el longlife learning.</li> </ul>			
<b>Bibliografía</b>				
<p>Moreno, J.C. (2014), Programación Orientada a Objetos. España: Rama</p> <p>Sznajdleder, P. (2017). Programación orientada a objetos y estructura de datos a fondo Implementación de algoritmos en Java, Alfa omega.</p> <p>Deitel, P. J.; Harvey, M. D. (2016). Cómo programar en java. México: Pearson educación.</p>				

## Sección IV. Desglose específico por cada unidad formativa

Tabla 4.6. Desglose específico de la unidad "Arreglos Unidimensionales y Multidimensionales."

<b>Número y nombre de la unidad:</b> 6. Arreglos Unidimensionales y Multidimensionales.							
<b>Tiempo y porcentaje para esta unidad:</b>		Teoría:	4 horas	Práctica:	4 horas	Porcentaje del programa:	8.89%
<b>Aprendizajes esperados:</b>		Reconocer el funcionamiento y sintaxis de los arreglos en la programación orientada a objetos para la solución de problemas laborales o cotidianos.					
Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad (Evidencia de aprendizaje de la unidad)			
6 Arreglos. 6.1 Unidimensional. 6.2 Multidimensional. 6.3. Objetos	<p>Saber:</p> <ul style="list-style-type: none"> <li>- Analizar los conceptos de Arreglos y sus tipos.</li> </ul> <p>Saber hacer:</p> <ul style="list-style-type: none"> <li>- Aplicar los diferentes tipos de arreglos en problemas matemáticos</li> </ul> <p>Ser:</p> <ul style="list-style-type: none"> <li>- Aporta puntos de vista con apertura a aprender de los otros y considera los de otras personas de manera reflexiva y respetuosa.</li> <li>- Participa activamente en la construcción de su aprendizaje y en la resolución de</li> </ul>	<ul style="list-style-type: none"> <li>-Exposición por parte del docente con apoyo de las presentaciones con los temas descritos.</li> <li>- Complementar información con material audiovisual.</li> <li>-Resolver el problemario de ejercicios para programar.</li> <li>- Realizar ejercicios programados para las actividades de cada tipo de arreglo.</li> </ul>	<ul style="list-style-type: none"> <li>- Evaluación Formativa: Realización de ejercicios prácticos. Instrumento: Rúbrica.</li> <li>- Evaluación sumativa: Elaborar el planteamiento de un problema laboral o cotidiano. Instrumento: Rúbrica o lista de cotejo.</li> </ul>	Planteamiento de un problema laboral o cotidiano en el que se pueda aplicar los arreglos que represente la solución en un programa con mayor complejidad.			



Continuación: Tabla 4.6. Desglose específico de la unidad "Arreglos Unidimensionales y Multidimensionales."

Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad
	<p>problemas, colaborando de manera productiva en espacios y equipos de trabajo.</p> <ul style="list-style-type: none"> <li>- Entrega en tiempo y forma sus actividades de aprendizaje siguiendo las indicaciones y considerando los criterios de evaluación.</li> <li>- Utiliza la tecnología para apoyar su aprendizaje y para el desarrollo de habilidades metacognitivas, el aprendizaje autónomo y el longlife learning.</li> </ul>			
<b>Bibliografía</b>				
<p>Moreno, J.C. (2014), Programación Orientada a Objetos. España: Rama</p> <p>Sznajdleder, P. (2017). Programación orientada a objetos y estructura de datos a fondo Implementación de algoritmos en Java, Alfa omega.</p> <p>Deitel, P. J.; Harvey, M. D. (2016). Cómo programar en java. México: Pearson educación.</p>				



## Sección IV. Desglose específico por cada unidad formativa

Tabla 4.7. Desglose específico de la unidad "Archivos binarios y de texto."

<b>Número y nombre de la unidad:</b> 7. Archivos binarios y de texto.							
<b>Tiempo y porcentaje para esta unidad:</b>		Teoría:	4 horas	Práctica:	2 horas	Porcentaje del programa:	6.67%
<b>Aprendizajes esperados:</b>		Diseñar programas con archivos y su aplicación en la programación orientada a objetos para el diseño de programas con requerimientos de almacenamiento permanente.					
Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad (Evidencia de aprendizaje de la unidad)			
<p>7 Archivos.</p> <p>7.1 Definición de Archivos de texto y archivos binarios.</p> <p>7.2 Operaciones básicas en archivos texto y binario.</p> <p>7.3 Manejo de excepciones en archivos.</p>	<p>Saber:</p> <ul style="list-style-type: none"> <li>- Analizar los conceptos de archivos y operaciones.</li> </ul> <p>Saber hacer:</p> <ul style="list-style-type: none"> <li>- Aplicar los archivos de texto y binarios en el diseño de programas que requieran almacenamiento de datos.</li> </ul> <p>Ser:</p> <ul style="list-style-type: none"> <li>- Aporta puntos de vista con apertura a aprender de los otros y considera los de otras personas de manera reflexiva y respetuosa.</li> <li>- Participa activamente en la construcción de su aprendizaje y en la resolución de</li> </ul>	<ul style="list-style-type: none"> <li>-Exposición por parte del docente con apoyo de las presentaciones con los temas descritos.</li> <li>- Complementar información con material audiovisual.</li> <li>-Resolver el problemario de ejercicios para programar.</li> <li>-Realizar ejercicios programados para las actividades acorde a los tipos de archivos.</li> </ul>	<ul style="list-style-type: none"> <li>- Evaluación Formativa: Realización de ejercicios prácticos. Instrumento: Rúbrica.</li> <li>- Evaluación sumativa: Elaborar el planteamiento de un problema laboral o cotidiano. Instrumento: Rúbrica o lista de cotejo.</li> </ul>	<p>Planteamiento de un problema laboral o cotidiano en el que se pueda aplicar los archivos que represente la solución en un programa.</p>			



Continuación: Tabla 4.7. Desglose específico de la unidad "Archivos binarios y de texto."

Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad
	<p>problemas, colaborando de manera productiva en espacios y equipos de trabajo.</p> <ul style="list-style-type: none"> <li>- Entrega en tiempo y forma sus actividades de aprendizaje siguiendo las indicaciones y considerando los criterios de evaluación.</li> <li>- Utiliza la tecnología para apoyar su aprendizaje y para el desarrollo de habilidades metacognitivas, el aprendizaje autónomo y el longlife learning.</li> </ul>			
<b>Bibliografía</b>				
<p>Moreno, J.C. (2014), Programación Orientada a Objetos. España: Rama</p> <p>Sznajdleder, P. (2017). Programación orientada a objetos y estructura de datos a fondo Implementación de algoritmos en Java, Alfa omega.</p> <p>Deitel, P. J.; Harvey, M. D. (2016). Cómo programar en java. México: Pearson educación.</p>				

## Sección IV. Desglose específico por cada unidad formativa

Tabla 4.8. Desglose específico de la unidad "Aplicaciones Gráficas."

<b>Número y nombre de la unidad:</b> 8. Aplicaciones Gráficas.				
<b>Tiempo y porcentaje para esta unidad:</b>		Teoría: 4 horas	Práctica: 2 horas	Porcentaje del programa: 6.67%
<b>Aprendizajes esperados:</b>		Diseñar programas con componentes, eventos, menús, gráficos e imágenes y su aplicación en la programación orientada a objetos para el diseño de programas con requerimientos de un problema cotidiano o laboral.		
Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad (Evidencia de aprendizaje de la unidad)
8 Aplicaciones Gráficas. 8.1 Principales Componentes y eventos. 8.2 Menús. 8.3 Gráficos e Imágenes. 8.4 Diseño e implementación de aplicaciones OO.	<b>Saber:</b> - Conocer las características y eventos de algunos componentes para la implementación de aplicaciones gráficas.  <b>Saber hacer:</b> - Diseñar programas de aplicaciones gráficas Orientado a Objetos.  <b>Ser:</b> - Aporta puntos de vista con apertura a aprender de los otros y considera los de otras personas de manera reflexiva y respetuosa.	-Exposición por parte del docente con apoyo de las presentaciones con los temas descritos. - Complementar información con material audiovisual. -Resolver el problemario de ejercicios para programar. -Realizar ejercicios programados para las actividades acorde a aplicaciones gráficas en OO.	<b>- Evaluación Formativa:</b> Realización de ejercicios prácticos. Instrumento: Rúbrica.  <b>- Evaluación sumativa:</b> Elaborar el planteamiento de un problema laboral o cotidiano. Instrumento: Rúbrica o lista de cotejo.	Planteamiento de un problema laboral o cotidiano en el que se pueda aplicar los archivos que represente la solución en un programa.



Continuación: Tabla 4.8. Desglose específico de la unidad "Aplicaciones Gráficas."

Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad
	<ul style="list-style-type: none"> <li>- Participa activamente en la construcción de su aprendizaje y en la resolución de problemas, colaborando de manera productiva en espacios y equipos de trabajo.</li> <li>- Entrega en tiempo y forma sus actividades de aprendizaje siguiendo las indicaciones y considerando los criterios de evaluación.</li> <li>- Utiliza la tecnología para apoyar su aprendizaje y para el desarrollo de habilidades metacognitivas, el aprendizaje autónomo y el longlife learning.</li> </ul>			
<b>Bibliografía</b>				
<p>Moreno, J.C. (2014), Programación Orientada a Objetos. España: Rama</p> <p>Sznajdleder, P. (2017). Programación orientada a objetos y estructura de datos a fondo Implementación de algoritmos en Java, Alfa omega.</p> <p>Deitel, P. J. ; Harvey, M. D. (2016). Cómo programar en java. México: Pearson educación.</p>				



## V. Perfil docente

Tabla 5. Descripción del perfil docente

Perfil deseable docente para impartir la asignatura
<p>Carrera(s): - Ingeniería en Tecnologías de software.</p> <ul style="list-style-type: none"><li>- Ingeniería en Informática.</li><li>- Ingeniería en computación.</li><li>- Ingeniería en Tecnología de la Información y Comunicación.</li><li>- Ingeniería en Tecnologías de la Información y Comunicación.</li><li>- Ingeniería en Tecnologías de la Información y Comunicaciones.</li><li>- Licenciatura en comunicación e información.</li><li>- Ingeniería en sistemas de computación.</li><li>- Ingeniería en sistemas computacionales.</li><li>- Licenciatura en Informática o Sistemas Computacionales o Maestría relacionada con el área de conocimiento.</li><li>- Licenciatura en ingeniería en sistemas computacionales.</li><li>- Licenciatura en ingeniería en sistemas de información.</li><li>- Ingeniería en sistemas estratégicos de información, especialización en redes.</li><li>- Licenciatura en sistemas computacionales.</li></ul>



o carrera afín

**Experiencia profesional:**

- Experiencia profesional comprobable correspondiente al campo disciplinar o carrera.
- Experiencia mínima de dos años.