

**DIRECCIÓN DE PLANIFICACIÓN Y DESARROLLO CURRICULAR  
MODALIDAD PRESENCIAL****PLAN DOCENTE DE LA ASIGNATURA****A. Datos básicos de la asignatura**

- **Nombre:** Programación Orientada a Objetos
- **Carrera:** Computación
- **Área Académica:** Técnica
- **Departamento:** Ciencias de la Computación y Electrónica
- **Sección departamental:** Tecnologías Avanzadas de la Web
- **Número de créditos:** 3
- **Campo de formación:**

Fundamentos Teóricos	<b>X</b>
Práxis Preprofesional	
Epistemología y metodología de la investigación	
Integración de contextos, saberes y cultura	
Comunicación y Lenguajes	

- **Unidad de organización curricular:**

Básica	Profesional	Integración curricular
X		

- **Período académico ordinario en el que se imparte:**  
Segundo
- **Periodo académico:** Abril - Agosto 2020
- **Pre-requisitos y co-requisitos:**

Los estudiantes de la asignatura de Programación Orientada a Objetos deben tener conocimientos sólidos previos en Fundamentos Computacionales, Fundamentos de la Programación, Estructuras Discretas, Fundamentos Matemáticos. En relación a programación la expertís del docente debe ser moderada o alta, en concordancia a temáticas como: elaboración de algoritmos, miniespecificaciones o pseudocódigo y manejo de algún lenguaje de programación de alto nivel.

- **Importancia de la asignatura en el perfil de egreso de la carrera:**

En el marco de la formación de un ingeniero en Computación, la asignatura tiene importancia, pues forma al estudiante con sólidas bases para la construcción de soluciones informáticas que permitan resolver problemas reales mediante el enfoque orientado a objetos, haciendo uso de la abstracción del mundo real, identificando las características esenciales de sus componentes y modelando su comportamiento; además permite la consolidación de bases teóricas y prácticas para el manejo del paradigma orientada a objetos mediante el uso de lenguajes de alto nivel y notación de lenguaje unificado.

- **Organización del aprendizaje:**

Organización del aprendizaje	
Componente	Número de horas
<b>Aprendizaje en contacto con el docente (incluye la hora de tutoría)</b>	48
<b>Aprendizaje práctico - experimental</b>	32
<b>Aprendizaje autónomo</b>	64
<b>Total:</b>	<b>144</b>

- **Horario de clases:**

Docente	Paralelo	Día	Aula	Horario
<b>René Rolando Elizalde Solano</b>	<b>A</b>	Lunes	Sala cómputo	Clases: 09:00 - 12:00
		Lunes	Sala cómputo	Práctica: 16:00 - 18:00

## B. Datos básicos del docente

- **Nombre:** René Rolando Elizalde Solano
- **Título:** Ingeniero en Sistemas
- **Departamento:** Ciencias de la Computación y Electrónica

- **Sección departamental:** Tecnologías Avanzadas de la WEB y SBC
- **Currículum resumido:**
  - Máster Universitario en Ingeniería del Software para la Web - Universidad de Alcalá. Ingeniero en Sistemas - Universidad Nacional de Loja. Docente Investigador Sección Tecnologías Avanzadas de la Web. Miembro del grupo de investigación Sistemas Basados en el Conocimiento Analista Desarrollador de la Sección de Desarrollo de Software de la Universidad Nacional de Loja. Docente invitado de la carrera de Ingeniería en Sistemas de la Universidad Nacional de Loja.

### C. Competencias a desarrollar

- **Competencias Genéricas de la UTPL:**
  - 1) Trabajo en Equipo
- **Competencias Específicas de la Carrera:**
  - 1) Demuestra dominio de conocimientos y habilidades para la solución de problemas mediante la aplicación y creación de algoritmos, modelos matemáticos y métodos de Ciencias de la Computación

### D. Planificación general de la asignatura

#### PRIMER BIMESTRE

Semana 1	
<b>Competencias de la carrera</b>	<ul style="list-style-type: none"> <li>Demuestra dominio de conocimientos y habilidades para la solución de problemas mediante la aplicación y creación de algoritmos, modelos matemáticos y métodos de Ciencias de la Computación.</li> </ul>
<b>Contenidos a desarrollarse</b>	Unidad 1: Conceptos Principales de Programación Orientada a Objetos <ul style="list-style-type: none"> <li>Introducción al Paradigma de Programación Orientada a Objetos</li> <li>Introducción al Lenguaje Unificado de Modelado</li> <li>Pilares fundamentales del Programación Orientada a Objetos: Abstracción, Polimorfismo, Herencia, Encapsulación</li> </ul>
<b>Resultados de aprendizaje</b>	<ul style="list-style-type: none"> <li>Diseña e implementa clases y objetos en soluciones computacionales.</li> </ul>
<b>Actividades del componente: Aprendizaje en</b>	<b>Actividades de aprendizaje asistido por el profesor:</b> <ul style="list-style-type: none"> <li>Clase magistral de contenidos</li> <li>Resolución de inquietudes</li> </ul>



<b>contacto con el docente</b>	<b>Actividades de aprendizaje colaborativo:</b> <ul style="list-style-type: none"><li>• Resolución de problemas</li><li>• Tutorías</li></ul>	
<b>Horas del componente: Aprendizaje en contacto con el docente</b>	3 horas	
<b>Actividades del componente: Aprendizaje práctico - experimental</b>	<ul style="list-style-type: none"><li>• Diagnóstico de conocimientos</li><li>• Talleres individuales y/o talleres grupales</li></ul>	
<b>Horas del componente: Aprendizaje práctico - experimental</b>	2 horas	
<b>Actividades del componente: Aprendizaje autónomo</b>	<ul style="list-style-type: none"><li>• Lectura de los temas de la Unidad 1 Conceptos Principales de Programación Orientada a Objetos:<ul style="list-style-type: none"><li>◦ Clases y Objetos</li><li>◦ Notación Lenguaje Unificado de Modelado (UML) para clases y objetos</li></ul>en los textos indicados en la bibliografía.</li><li>• Búsqueda de información de la temática de la unidad 1: Notación Lenguaje Unificado de Modelado (UML) para clases y objetos en fuentes web académicas - (Consulta de temáticas)</li><li>• Elaboración de ejercicios o preguntas de la unidad I - (Talleres individuales Extraclase)</li></ul>	<ul style="list-style-type: none"><li>• 1 hora</li><li>• 1 horas</li><li>• 2 horas</li></ul>
<b>Horas del componente: Aprendizaje autónomo</b>	4 horas	

<b>Semana 2</b>		
<b>Competencias de la carrera</b>	<ul style="list-style-type: none"> <li>Demuestra dominio de conocimientos y habilidades para la solución de problemas mediante la aplicación y creación de algoritmos, modelos matemáticos y métodos de Ciencias de la Computación.</li> </ul>	
<b>Contenidos a desarrollarse</b>	Unidad 1: Conceptos Principales de Programación Orientada a Objetos <ul style="list-style-type: none"> <li>Clases y Objetos</li> <li>Notación Lenguaje Unificado de Modelado (UML) para clases y objetos</li> </ul>	
<b>Resultados de aprendizaje</b>	<ul style="list-style-type: none"> <li>Aplica conceptos de UML en el desarrollo de soluciones computacionales.</li> </ul>	
<b>Actividades del componente: Aprendizaje en contacto con el docente</b>	<b>Actividades de aprendizaje asistido por el profesor:</b> <ul style="list-style-type: none"> <li>Clase magistral de contenidos</li> <li>Resolución de inquietudes</li> </ul> <b>Actividades de aprendizaje colaborativo:</b> <ul style="list-style-type: none"> <li>Resolución de problemas</li> <li>Tutorías</li> </ul>	
<b>Horas del componente: Aprendizaje en contacto con el docente</b>	3 horas	
<b>Actividades del componente: Aprendizaje práctico - experimental</b>	<ul style="list-style-type: none"> <li>Talleres individuales y/o talleres grupales</li> </ul>	
<b>Horas del componente: Aprendizaje práctico - experimental</b>	2 horas	
<b>Actividades del componente: Aprendizaje autónomo</b>	<ul style="list-style-type: none"> <li>Lectura de los temas relacionados a la Unidad 2: Estructura y creación de programas en Programación Orientada a Objetos:               <ul style="list-style-type: none"> <li>Manipulación de métodos en Programación Orientada a Objetos</li> <li>Declaración de clases en lenguajes de alto nivel</li> <li>Creación y manipulación de objetos en lenguajes de alto nivel</li> </ul> </li> <li>Uso de métodos establecer y obtener en los textos indicados en la bibliografía</li> <li>Búsqueda de información de las temáticas de la Unidad 2: Creación y manipulación de objetos en lenguajes de alto nivel; Uso de métodos establecer</li> </ul>	<ul style="list-style-type: none"> <li>1 horas</li> <li>2 horas</li> </ul>



	<p>y obtener en fuentes web académicas - (Consulta de temáticas)</p> <ul style="list-style-type: none"><li>• Elaboración de ejercicios o preguntas de la unidad 1 - (Talleres individuales Extraclase)</li></ul>	
<b>Horas del componente: Aprendizaje autónomo</b>	4 horas	

<b>Semana 3</b>		
<b>Competencias de la carrera</b>	<ul style="list-style-type: none"> <li>• Demuestra dominio de conocimientos y habilidades para la solución de problemas mediante la aplicación y creación de algoritmos, modelos matemáticos y métodos de Ciencias de la Computación.</li> </ul>	
<b>Contenidos a desarrollarse</b>	Unidad 2: Estructura y creación de programas en Programación Orientada a Objetos <ul style="list-style-type: none"> <li>• Manipulación de métodos en Programación Orientada a Objetos</li> <li>• Declaración de clases en lenguajes de alto nivel</li> <li>• Creación y manipulación de objetos en lenguajes de alto nivel</li> <li>• Uso de métodos establecer y obtener</li> </ul>	
<b>Resultados de aprendizaje</b>	<ul style="list-style-type: none"> <li>• Explica e implementa relaciones de clases en el ámbito de Programación Orientado a Objetos.</li> <li>• Aplica conceptos de UML en el desarrollo de soluciones computacionales.</li> </ul>	
<b>Actividades del componente: Aprendizaje en contacto con el docente</b>	<b>Actividades de aprendizaje asistido por el profesor:</b> <ul style="list-style-type: none"> <li>• Clase magistral de contenidos</li> <li>• Resolución de inquietudes</li> </ul> <b>Actividades de aprendizaje colaborativo:</b> <ul style="list-style-type: none"> <li>• Resolución de problemas</li> <li>• Tutorías</li> </ul>	
<b>Horas del componente: Aprendizaje en contacto con el docente</b>	3 horas	
<b>Actividades del componente: Aprendizaje práctico - experimental</b>	<ul style="list-style-type: none"> <li>• Talleres individuales y/o talleres grupales</li> </ul>	
<b>Horas del componente: Aprendizaje práctico - experimental</b>	2 horas	
<b>Actividades del componente: Aprendizaje autónomo</b>	<ul style="list-style-type: none"> <li>• Lectura de los temas pendientes de estudio de la Unidad 2: Estructura y creación de programas en Programación Orientada a Objetos               <ul style="list-style-type: none"> <li>◦ Manejo de constructores</li> <li>◦ Relación entre clases: herencia, agregación, composición y agregación en los textos indicados en la bibliografía.</li> </ul> </li> <li>• Elaboración de ejercicios de la unidad 2 - (Talleres individuales Extraclase)</li> <li>• Proyecto Bimestral – fase 1</li> </ul>	<ul style="list-style-type: none"> <li>• 2 horas</li> <li>• 1 hora</li> <li>• 1 hora</li> </ul>
<b>Horas del componente:</b>	4 horas	



**UTPL**  
UNIVERSIDAD TÉCNICA PARTICULAR DE LIBIA

**VICERRECTORADO ACADÉMICO**

**Aprendizaje  
autónomo**



<b>Semana 4</b>		
<b>Competencias de la carrera</b>	<ul style="list-style-type: none"> <li>• Demuestra dominio de conocimientos y habilidades para la solución de problemas mediante la aplicación y creación de algoritmos, modelos matemáticos y métodos de Ciencias de la Computación.</li> </ul>	
<b>Contenidos a desarrollarse</b>	Unidad 2: Estructura y creación de programas en Programación Orientada a Objetos <ul style="list-style-type: none"> <li>• Manejo de constructores</li> <li>• Relación entre clases: herencia, agregación, composición y agregación</li> </ul>	
<b>Resultados de aprendizaje</b>	<ul style="list-style-type: none"> <li>• Explica e implementa relaciones de clases en el ámbito de Programación Orientado a Objetos.</li> <li>• Aplica conceptos de UML en el desarrollo de soluciones computacionales.</li> </ul>	
<b>Actividades del componente: Aprendizaje en contacto con el docente</b>	<b>Actividades de aprendizaje asistido por el profesor:</b> <ul style="list-style-type: none"> <li>• Clase magistral de contenidos</li> <li>• Resolución de inquietudes</li> </ul> <b>Actividades de aprendizaje colaborativo:</b> <ul style="list-style-type: none"> <li>• Resolución de problemas</li> <li>• Tutorías</li> </ul>	
<b>Horas del componente: Aprendizaje en contacto con el docente</b>	3 horas	
<b>Actividades del componente: Aprendizaje práctico - experimental</b>	<ul style="list-style-type: none"> <li>• Talleres individuales y/o talleres grupales</li> </ul>	
<b>Horas del componente: Aprendizaje práctico - experimental</b>	2 horas	
<b>Actividades del componente: Aprendizaje autónomo</b>	<ul style="list-style-type: none"> <li>• Lectura de los temas relacionados a la Unidad 3: Manejo de Estructuras               <ul style="list-style-type: none"> <li>◦ Estructura de control en Programación Orientada a Objetos (condicionales y repetitivas) en los textos indicados en la bibliografía.</li> </ul> </li> <li>• Búsqueda de información sobre la Unidad 3 en el ámbito de estructuras repetitivas en fuentes web académicas - (Consulta de temáticas).</li> <li>• Elaboración de ejercicios de la unidad 2 - (Talleres individuales Extraclase).</li> </ul>	<ul style="list-style-type: none"> <li>• 1 hora</li> <li>• 1 hora</li> <li>• 2 horas</li> </ul>
<b>Horas del componente: Aprendizaje autónomo</b>	4 horas	



**Semana 5**

<b>Competencias de la carrera</b>	<ul style="list-style-type: none"> <li>Demuestra dominio de conocimientos y habilidades para la solución de problemas mediante la aplicación y creación de algoritmos, modelos matemáticos y métodos de Ciencias de la Computación.</li> </ul>	
<b>Contenidos a desarrollarse</b>	Unidad 3: Manejo de Estructuras <ul style="list-style-type: none"> <li>Estructura de control en Programación Orientada a Objetos (condicionales y repetitivas)</li> </ul>	
<b>Resultados de aprendizaje</b>	<ul style="list-style-type: none"> <li>Diseña e implementa clases y objetos en soluciones computacionales.</li> <li>Explica e implementa relaciones de clases en el ámbito de Programación Orientada a Objetos.</li> <li>Aplica conceptos de UML en el desarrollo de soluciones computacionales.</li> </ul>	
<b>Actividades del componente: Aprendizaje en contacto con el docente</b>	<b>Actividades de aprendizaje asistido por el profesor:</b> <ul style="list-style-type: none"> <li>Clase magistral de contenidos</li> <li>Resolución de inquietudes</li> </ul> <b>Actividades de aprendizaje colaborativo:</b> <ul style="list-style-type: none"> <li>Resolución de problemas</li> <li>Tutorías</li> </ul>	
<b>Horas del componente: Aprendizaje en contacto con el docente</b>	3 horas	
<b>Actividades del componente: Aprendizaje práctico - experimental</b>	<ul style="list-style-type: none"> <li>Talleres individuales y/o talleres grupales</li> <li>Evaluación parcial del primer bimestre</li> </ul>	
<b>Horas del componente: Aprendizaje práctico - experimental</b>	2 horas	
<b>Actividades del componente: Aprendizaje autónomo</b>	<ul style="list-style-type: none"> <li>Lectura de los temas relacionados a la Unidad 3: Manejo de Estructuras               <ul style="list-style-type: none"> <li>Estructuras de datos en Programación Orientada a Objetos: Manejo de estructuras estáticas en los textos indicados en la bibliografía</li> </ul> </li> <li>Elaboración de ejercicios sobre Estructura de control en Programación Orientada a Objetos (condicionales y repetitivas) - (Talleres individuales Extraclase)</li> </ul>	<ul style="list-style-type: none"> <li>2 horas</li> <li>2 horas</li> </ul>
<b>Horas del componente: Aprendizaje autónomo</b>	4 horas	

<b>Semana 6</b>		
<b>Competencias de la carrera</b>	<ul style="list-style-type: none"> <li>Demuestra dominio de conocimientos y habilidades para la solución de problemas mediante la aplicación y creación de algoritmos, modelos matemáticos y métodos de Ciencias de la Computación.</li> </ul>	
<b>Contenidos a desarrollarse</b>	Unidad 3: Manejo de Estructuras <ul style="list-style-type: none"> <li>Estructuras de datos en Programación Orientada a Objetos               <ul style="list-style-type: none"> <li>Manejo de estructuras estáticas (arreglos de objetos))</li> </ul> </li> </ul>	
<b>Resultados de aprendizaje</b>	<ul style="list-style-type: none"> <li>Diseña e implementa clases y objetos en soluciones computacionales.</li> <li>Explica e implementa relaciones de clases en el ámbito de Programación Orientada a Objetos.</li> <li>Define y usa estructuras de datos de tipo no primitivas.</li> <li>Aplica conceptos de UML en el desarrollo de soluciones computacionales.</li> </ul>	
<b>Actividades del componente: Aprendizaje en contacto con el docente</b>	<b>Actividades de aprendizaje asistido por el profesor:</b> <ul style="list-style-type: none"> <li>Clase magistral de contenidos</li> <li>Resolución de inquietudes</li> </ul> <b>Actividades de aprendizaje colaborativo:</b> <ul style="list-style-type: none"> <li>Resolución de problemas</li> <li>Tutorías</li> </ul>	
<b>Horas del componente: Aprendizaje en contacto con el docente</b>	3 horas	
<b>Actividades del componente: Aprendizaje práctico - experimental</b>	<ul style="list-style-type: none"> <li>Talleres individuales y/o talleres grupales</li> </ul>	
<b>Horas del componente: Aprendizaje práctico - experimental</b>	2 horas	
<b>Actividades del componente: Aprendizaje autónomo</b>	<ul style="list-style-type: none"> <li>Lectura de los temas relacionados a               <ul style="list-style-type: none"> <li>Unidad 3: Manejo de Estructuras                   <ul style="list-style-type: none"> <li>Estructuras de datos en Programación Orientada a Objetos: Manejo de estructuras dinámicas</li> <li>Serialización de objetos</li> </ul> </li> </ul>               en los textos indicados en la bibliografía             </li> <li>Elaboración de ejercicios de Estructuras de datos en Programación Orientada a Objetos: Manejo de estructuras estáticas - (Talleres individuales Extraclase)</li> </ul>	<ul style="list-style-type: none"> <li>2 horas</li> <li>1 hora</li> <li>1 hora</li> </ul>



**UTPL**  
UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA

## VICERRECTORADO ACADÉMICO

	<ul style="list-style-type: none"><li>• Proyecto Bimestral – fase 2</li></ul>	
<b>Horas del componente: Aprendizaje autónomo</b>	4 horas	

<b>Semana 7</b>		
<b>Competencias de la carrera</b>	<ul style="list-style-type: none"> <li>• Demuestra dominio de conocimientos y habilidades para la solución de problemas mediante la aplicación y creación de algoritmos, modelos matemáticos y métodos de Ciencias de la Computación.</li> </ul>	
<b>Contenidos a desarrollarse</b>	Unidad 3: Manejo de Estructuras <ul style="list-style-type: none"> <li>• Estructuras de datos en Programación Orientada a Objetos               <ul style="list-style-type: none"> <li>◦ Manejo de estructuras dinámicas (ArrayList y Colecciones de objetos)</li> </ul> </li> <li>• Serialización de objetos</li> </ul>	
<b>Resultados de aprendizaje</b>	<ul style="list-style-type: none"> <li>• Diseña e implementa clases y objetos en soluciones computacionales.</li> <li>• Explica e implementa relaciones de clases en el ámbito de Programación Orientada a Objetos.</li> <li>• Define y usa estructuras de datos de tipo no primitivas.</li> <li>• Aplica conceptos de UML en el desarrollo de soluciones computacionales.</li> </ul>	
<b>Actividades del componente: Aprendizaje en contacto con el docente</b>	<b>Actividades de aprendizaje asistido por el profesor:</b> <ul style="list-style-type: none"> <li>• Clase magistral de contenidos</li> <li>• Resolución de inquietudes</li> </ul> <b>Actividades de aprendizaje colaborativo:</b> <ul style="list-style-type: none"> <li>• Resolución de problemas</li> <li>• Tutorías</li> </ul>	
<b>Horas del componente: Aprendizaje en contacto con el docente</b>	3 horas	
<b>Actividades del componente: Aprendizaje práctico - experimental</b>	<ul style="list-style-type: none"> <li>• Talleres individuales y/o talleres grupales</li> </ul>	
<b>Horas del componente: Aprendizaje práctico - experimental</b>	2 horas	
<b>Actividades del componente: Aprendizaje autónomo</b>	<ul style="list-style-type: none"> <li>• Revisión de contenidos de la unidad 1,2,3; como preparación del evaluación del final del primer bimestre.</li> </ul>	<ul style="list-style-type: none"> <li>• 4 horas</li> </ul>
<b>Horas del componente: Aprendizaje autónomo</b>	4 horas	



**Semana 8**

<b>Competencias de la carrera</b>	<ul style="list-style-type: none"> <li>Demuestra dominio de conocimientos y habilidades para la solución de problemas mediante la aplicación y creación de algoritmos, modelos matemáticos y métodos de Ciencias de la Computación.</li> </ul>	
<b>Contenidos a desarrollarse</b>	<b>Evaluación del primer bimestre</b>	
<b>Resultados de aprendizaje</b>	<ul style="list-style-type: none"> <li>Diseña e implementa clases y objetos en soluciones computacionales.</li> <li>Aplica conceptos de UML en el desarrollo de soluciones computacionales.</li> <li>Explica e implementa relaciones de clases en el ámbito de Programación Orientado a Objetos.</li> <li>Define y usa estructuras de datos de tipo no primitivas.</li> </ul>	
<b>Actividades del componente: Aprendizaje en contacto con el docente</b>	<b>Actividades de aprendizaje asistido por el profesor:</b> <ul style="list-style-type: none"> <li>Explicación y resolución de inquietudes durante evaluación bimestral</li> </ul>	
<b>Horas del componente: Aprendizaje en contacto con el docente</b>	3 horas	
<b>Actividades del componente: Aprendizaje práctico - experimental</b>	<ul style="list-style-type: none"> <li>Desarrollo guiado de ejercicios del primer bimestre</li> </ul>	
<b>Horas del componente: Aprendizaje práctico - experimental</b>	2 horas	
<b>Actividades del componente: Aprendizaje autónomo</b>	<ul style="list-style-type: none"> <li>Lectura de los temas de la Unidad 4: Manejo de Herencia               <ul style="list-style-type: none"> <li>Conceptos generales: superclases, subclases, niveles de acceso</li> <li>Jerarquía de Herencia en Diagramas UML,</li> </ul> </li> <li>en los textos indicados en la bibliografía.</li> <li>Búsqueda de información sobre Jerarquía de Herencia en Diagramas UML en fuentes web académicas - (Consulta de temáticas)</li> <li>Proyecto Bimestral - entrega final</li> </ul>	<ul style="list-style-type: none"> <li>2 horas</li> <li>1 hora</li> <li>1 hora</li> </ul>
<b>Horas del componente: Aprendizaje autónomo</b>	4 horas	

**Horas de trabajo:** (Totales del bimestre)

**Componente de aprendizaje en contacto con el docente:** 24 horas

**Componente de aprendizaje práctico - experimental:** 16 horas

**Componente de aprendizaje autónomo:** 32 horas

**Fechas importantes:**

- Semana 1: inicio de actividades
- Semana 5: evaluación parcial del segundo bimestre
- Semana 8: evaluación final del segundo bimestre

## Evaluación del primer bimestre:

Componente	%	Actividad	Instrumento de evaluación	Criterios de Evaluación	Tipo de evaluación	Modalidad evaluación	Ponderación
Aprendizaje en contacto con el docente (3.5 puntos)	35%	Evaluación inicial	Prueba teórica práctica	El estudiante en formación puede: <ul style="list-style-type: none"><li>Diseñar clases y objetos en soluciones computacionales.</li><li>Implementar clases y objetos en soluciones computacionales a través de lenguajes de alto nivel.</li><li>Estructurar y crear programas bajo el paradigma de Programación Orientada a Objetos.</li><li>Definir y usar estructuras de control y datos en Programación Orientada a Objetos.</li></ul>	Diagnóstica	Heteroevaluación	0 puntos
		Evaluación parcial	Prueba teórica práctica		Sumativa	Heteroevaluación	10%
		Evaluación bimestral	Prueba teórica práctica		Sumativa	Heteroevaluación	25%
		SUBTOTAL					35%
	Aprendizaje práctico - experimental (3.5 puntos)	35%	Talleres Individuales	Resolución de ejercicios	El estudiante en formación puede: <ul style="list-style-type: none"><li>Diseñar clases y objetos en soluciones computacionales.</li><li>Implementar clases y objetos en soluciones computacionales a través de lenguajes de alto nivel.</li><li>Estructurar y crear programas bajo el paradigma de Programación Orientada a Objetos.</li><li>Definir y usar estructuras de control y datos en Programación Orientada a Objetos.</li></ul>	Sumativa	Heteroevaluación
Talleres Grupales			Resolución de ejercicios	Sumativa		Heteroevaluación	15%
SUBTOTAL						35%	
Aprendizaje autónomo (3 puntos)		30%	Proyecto bimestral	Resolución de problemas	El estudiante en formación puede: <ul style="list-style-type: none"><li>Diseñar clases y objetos en soluciones computacionales.</li><li>Implementar clases y objetos en soluciones</li></ul>	Sumativa	Heteroevaluación
	Talleres		Resolución	Sumativa		Heteroevaluación	10%



		individuales Extraclase	de ejercicios				
		Consulta de temáticas	Resúmenes / Investigación		<ul style="list-style-type: none"><li>• computacionales a través de lenguajes de alto nivel. Estructurar y crear programas bajo el paradigma de Programación Orientada a Objetos.</li></ul>	Sumativa	Heteroevaluación
		SUBTOTAL					30 %

TOTAL	10 puntos
-------	-----------

**SEGUNDO BIMESTRE**

<b>Semana 1</b>		
<b>Competencias de la carrera</b>	<ul style="list-style-type: none"> <li>Demuestra dominio de conocimientos y habilidades para la solución de problemas mediante la aplicación y creación de algoritmos, modelos matemáticos y métodos de Ciencias de la Computación.</li> </ul>	
<b>Contenidos a desarrollarse</b>	Unidad 4: Manejo de Herencia <ul style="list-style-type: none"> <li>Conceptos generales: superclases, subclases, niveles de acceso</li> <li>Jerarquía de Herencia en Diagramas UML</li> </ul>	
<b>Resultados de aprendizaje</b>	<ul style="list-style-type: none"> <li>Explica e implementa relaciones de clases en el ámbito de Programación Orientado a Objetos.</li> <li>Usa subclases para diseñar una jerarquía de clases.</li> <li>Aplica conceptos de UML en el desarrollo de soluciones computacionales.</li> </ul>	
<b>Actividades del componente: Aprendizaje en contacto con el docente</b>	<b>Actividades de aprendizaje asistido por el profesor:</b> <ul style="list-style-type: none"> <li>Clase magistral de contenidos</li> <li>Resolución de inquietudes</li> </ul> <b>Actividades de aprendizaje colaborativo:</b> <ul style="list-style-type: none"> <li>Resolución de problemas</li> <li>Tutorías</li> </ul>	
<b>Horas del componente: Aprendizaje en contacto con el docente</b>	3 horas	
<b>Actividades del componente: Aprendizaje práctico - experimental</b>	<ul style="list-style-type: none"> <li>Talleres individuales y/o talleres grupales</li> </ul>	
<b>Horas del componente: Aprendizaje práctico - experimental</b>	2 horas	
<b>Actividades del componente: Aprendizaje autónomo</b>	<ul style="list-style-type: none"> <li>Lectura de los temas pendientes de estudio de la Unidad 4: Manejo de Herencia               <ul style="list-style-type: none"> <li>Invocación de constructores</li> <li>Implementación de Herencia en lenguaje de alto nivel</li> </ul> </li> <li>en los textos indicados en la bibliografía</li> <li>Elaboración de ejercicios sobre Jerarquía de Herencia en Diagramas UML - (Talleres individuales Extraclase).</li> </ul>	<ul style="list-style-type: none"> <li>2 horas</li> <li>2 horas</li> </ul>
<b>Horas del componente: Aprendizaje autónomo</b>	4 horas	

<b>Semana 2</b>		
<b>Competencias de la carrera</b>	<ul style="list-style-type: none"> <li>• Demuestra dominio de conocimientos y habilidades para la solución de problemas mediante la aplicación y creación de algoritmos, modelos matemáticos y métodos de Ciencias de la Computación.</li> </ul>	
<b>Contenidos a desarrollarse</b>	Unidad 4: Manejo de Herencia <ul style="list-style-type: none"> <li>• Invocación de constructores</li> <li>• Implementación de Herencia en lenguaje de alto nivel</li> </ul>	
<b>Resultados de aprendizaje</b>	<ul style="list-style-type: none"> <li>• Explica e implementa relaciones de clases en el ámbito de Programación Orientado a Objetos.</li> <li>• Usa subclases para diseñar una jerarquía de clases.</li> <li>• Aplica conceptos de UML en el desarrollo de soluciones computacionales.</li> </ul>	
<b>Actividades del componente: Aprendizaje en contacto con el docente</b>	<b>Actividades de aprendizaje asistido por el profesor:</b> <ul style="list-style-type: none"> <li>• Clase magistral de contenidos</li> <li>• Resolución de inquietudes</li> </ul> <b>Actividades de aprendizaje colaborativo:</b> <ul style="list-style-type: none"> <li>• Resolución de problemas</li> <li>• Tutorías</li> </ul>	
<b>Horas del componente: Aprendizaje en contacto con el docente</b>	3 horas	
<b>Actividades del componente: Aprendizaje práctico - experimental</b>	<ul style="list-style-type: none"> <li>• Talleres individuales y/o talleres grupales</li> </ul>	
<b>Horas del componente: Aprendizaje práctico - experimental</b>	2 horas	
<b>Actividades del componente: Aprendizaje autónomo</b>	<ul style="list-style-type: none"> <li>• Lectura de los temas de estudio Unidad 5: Implementación de Polimorfismo               <ul style="list-style-type: none"> <li>◦ Conceptos generales: clases y métodos abstractos</li> <li>◦ Diseño de Diagramas UML bajo el Polimorfismo</li> </ul>               en los textos indicados en la bibliografía.             </li> <li>• Elaboración de ejercicios relacionados a Implementación de Herencia en lenguaje de alto nivel - (Talleres individuales Extraclase)</li> </ul>	<ul style="list-style-type: none"> <li>• 2 horas</li> <li>• 2 horas</li> </ul>
<b>Horas del componente: Aprendizaje autónomo</b>	4 horas	

<b>Semana 3</b>		
<b>Competencias de la carrera</b>	<ul style="list-style-type: none"> <li>• Demuestra dominio de conocimientos y habilidades para la solución de problemas mediante la aplicación y creación de algoritmos, modelos matemáticos y métodos de Ciencias de la Computación.</li> </ul>	
<b>Contenidos a desarrollarse</b>	Unidad 5: Implementación de Polimorfismo <ul style="list-style-type: none"> <li>• Conceptos generales: clases y métodos abstractos</li> <li>• Diseño de Diagramas UML bajo el polimorfismo</li> </ul>	
<b>Resultados de aprendizaje</b>	<ul style="list-style-type: none"> <li>• Explica e implementa relaciones de clases en el ámbito de Programación Orientado a Objetos.</li> <li>• Usa subclasses para diseñar una jerarquía de clases.</li> <li>• Aplica conceptos de UML en el desarrollo de soluciones computacionales.</li> </ul>	
<b>Actividades del componente: Aprendizaje en contacto con el docente</b>	<b>Actividades de aprendizaje asistido por el profesor:</b> <ul style="list-style-type: none"> <li>• Clase magistral de contenidos</li> <li>• Resolución de inquietudes</li> </ul> <b>Actividades de aprendizaje colaborativo:</b> <ul style="list-style-type: none"> <li>• Resolución de problemas</li> <li>• Tutorías</li> </ul>	
<b>Horas del componente: Aprendizaje en contacto con el docente</b>	3 horas	
<b>Actividades del componente: Aprendizaje práctico - experimental</b>	<ul style="list-style-type: none"> <li>• Talleres individuales y/o talleres grupales</li> </ul>	
<b>Horas del componente: Aprendizaje práctico - experimental</b>	2 horas	
<b>Actividades del componente: Aprendizaje autónomo</b>	<ul style="list-style-type: none"> <li>• Lectura de los temas de la Unidad 5: Implementación de Polimorfismo:               <ul style="list-style-type: none"> <li>◦ Implementación de Polimorfismo en lenguaje de alto nivel en los textos indicados en la bibliografía.</li> </ul> </li> <li>• Búsqueda de información: sobre Implementación de Polimorfismo en lenguaje de alto nivel en fuentes web académicas - (Consulta de temáticas).</li> <li>• Elaboración de ejercicios sobre Diseño de Diagramas UML bajo el polimorfismo - (Talleres individuales Extraclase)</li> <li>• Proyecto Bimestral - fase 1</li> </ul>	<ul style="list-style-type: none"> <li>• 1 hora</li> <li>• 1 hora</li> <li>• 1 hora</li> <li>• 1 hora</li> </ul>
<b>Horas del componente: Aprendizaje</b>	4 horas	



VICERRECTORADO ACADÉMICO

autónomo

<b>Semana 4</b>		
<b>Competencias de la carrera</b>	<ul style="list-style-type: none"> <li>• Demuestra dominio de conocimientos y habilidades para la solución de problemas mediante la aplicación y creación de algoritmos, modelos matemáticos y métodos de Ciencias de la Computación.</li> </ul>	
<b>Contenidos a desarrollarse</b>	Unidad 5: Implementación de Polimorfismo <ul style="list-style-type: none"> <li>• Implementación de Polimorfismo en lenguaje de alto nivel.</li> </ul>	
<b>Resultados de aprendizaje</b>	<ul style="list-style-type: none"> <li>• Explica e implementa relaciones de clases en el ámbito de Programación Orientado a Objetos.</li> <li>• Usa subclases para diseñar una jerarquía de clases.</li> <li>• Aplica conceptos de UML en el desarrollo de soluciones computacionales.</li> </ul>	
<b>Actividades del componente: Aprendizaje en contacto con el docente</b>	<b>Actividades de aprendizaje asistido por el profesor:</b> <ul style="list-style-type: none"> <li>• Clase magistral de contenidos</li> <li>• Resolución de inquietudes</li> </ul> <b>Actividades de aprendizaje colaborativo:</b> <ul style="list-style-type: none"> <li>• Resolución de problemas</li> <li>• Tutorías</li> </ul>	
<b>Horas del componente: Aprendizaje en contacto con el docente</b>	3 horas	
<b>Actividades del componente: Aprendizaje práctico - experimental</b>	<ul style="list-style-type: none"> <li>• Talleres individuales y/o talleres grupales</li> </ul>	
<b>Horas del componente: Aprendizaje práctico - experimental</b>	2 horas	
<b>Actividades del componente: Aprendizaje autónomo</b>	<ul style="list-style-type: none"> <li>• Lectura de los temas de la Unidad 6: Desarrollo basado en pruebas en Programación Orientada a Objetos:               <ul style="list-style-type: none"> <li>◦ Uso de librerías para la optimización de pruebas en textos indicados en la bibliografía.</li> </ul> </li> <li>• Elaboración de ejercicios sobre implementación de Polimorfismo en lenguaje de alto nivel (Talleres individuales Extraclase).</li> </ul>	<ul style="list-style-type: none"> <li>• 2 horas</li> <li>• 2 hora</li> </ul>
<b>Horas del componente: Aprendizaje autónomo</b>	Total de horas	

<b>Semana 5</b>		
<b>Competencias de la carrera</b>	<ul style="list-style-type: none"> <li>• Demuestra dominio de conocimientos y habilidades para la solución de problemas mediante la aplicación y creación de algoritmos, modelos matemáticos y métodos de Ciencias de la Computación.</li> </ul>	
<b>Contenidos a desarrollarse</b>	Unidad 6: Desarrollo basado en pruebas en Programación Orientada a Objetos <ul style="list-style-type: none"> <li>• Uso de librerías para la optimización de pruebas</li> </ul>	
<b>Resultados de aprendizaje</b>	<ul style="list-style-type: none"> <li>• Explica e implementa relaciones de clases en el ámbito de Programación Orientado a Objetos.</li> <li>• Usa subclases para diseñar una jerarquía de clases.</li> <li>• Aplica conceptos de UML en el desarrollo de soluciones computacionales.</li> </ul>	
<b>Actividades del componente: Aprendizaje en contacto con el docente</b>	<b>Actividades de aprendizaje asistido por el profesor:</b> <ul style="list-style-type: none"> <li>• Clase magistral de contenidos</li> <li>• Resolución de inquietudes</li> <li>• Evaluación del parcial del segundo bimestre</li> </ul> <b>Actividades de aprendizaje colaborativo:</b> <ul style="list-style-type: none"> <li>• Resolución de problemas</li> <li>• Tutorías</li> </ul>	
<b>Horas del componente: Aprendizaje en contacto con el docente</b>	3 horas	
<b>Actividades del componente: Aprendizaje práctico - experimental</b>	<ul style="list-style-type: none"> <li>• Talleres individuales y/o talleres grupales</li> </ul>	
<b>Horas del componente: Aprendizaje práctico - experimental</b>	2 horas	
<b>Actividades del componente: Aprendizaje autónomo</b>	<ul style="list-style-type: none"> <li>• Lectura de los temas de la Unidad 7: Principios SOLID               <ul style="list-style-type: none"> <li>◦ Single responsibility</li> <li>◦ Open-closed</li> <li>◦ Liskov substitution</li> </ul> </li> </ul> <p>en los textos indicados en la bibliografía.</p> <ul style="list-style-type: none"> <li>• Elaboración de ejercicios sobre Desarrollo basado en pruebas en Programación Orientada a Objetos - (Talleres individuales Extraclase)</li> </ul>	<ul style="list-style-type: none"> <li>• 2 hora</li> <li>• 2 hora</li> </ul>
<b>Horas del componente: Aprendizaje autónomo</b>	4 horas	

<b>Semana 6</b>		
<b>Competencias de la carrera</b>	<ul style="list-style-type: none"> <li>• Demuestra dominio de conocimientos y habilidades para la solución de problemas mediante la aplicación y creación de algoritmos, modelos matemáticos y métodos de Ciencias de la Computación.</li> </ul>	
<b>Contenidos a desarrollarse</b>	Unidad 7: Principios SOLID <ul style="list-style-type: none"> <li>• Single responsibility</li> <li>• Open-closed</li> <li>• Liskov substitution</li> </ul>	
<b>Resultados de aprendizaje</b>	<ul style="list-style-type: none"> <li>• Explica e implementa relaciones de clases en el ámbito de Programación Orientado a Objetos.</li> <li>• Usa subclases para diseñar una jerarquía de clases.</li> <li>• Aplica conceptos de UML en el desarrollo de soluciones computacionales.</li> </ul>	
<b>Actividades del componente: Aprendizaje en contacto con el docente</b>	<b>Actividades de aprendizaje asistido por el profesor:</b> <ul style="list-style-type: none"> <li>• Clase magistral de contenidos</li> <li>• Resolución de inquietudes</li> </ul> <b>Actividades de aprendizaje colaborativo:</b> <ul style="list-style-type: none"> <li>• Resolución de problemas</li> <li>• Tutorías</li> </ul>	
<b>Horas del componente: Aprendizaje en contacto con el docente</b>	3 horas	
<b>Actividades del componente: Aprendizaje práctico - experimental</b>	<ul style="list-style-type: none"> <li>• Talleres individuales y/o talleres grupales</li> </ul>	
<b>Horas del componente: Aprendizaje práctico - experimental</b>	2 horas	
<b>Actividades del componente: Aprendizaje autónomo</b>	<ul style="list-style-type: none"> <li>• Lectura de los temas de la Unidad 7: Principios SOLID               <ul style="list-style-type: none"> <li>◦ Interface segregation</li> <li>◦ Dependency inversion</li> </ul> </li> <li>en los textos indicados en la bibliografía.</li> <li>• Elaboración de ejercicios sobre Principios SOLID - (Talleres individuales Extraclase)</li> <li>• Proyecto Bimestral - fase 2</li> </ul>	<ul style="list-style-type: none"> <li>• 2 horas</li> <li>• 1 hora</li> <li>• 1 hora</li> </ul>
<b>Horas del componente: Aprendizaje autónomo</b>	4 horas	



<b>Semana 7</b>		
<b>Competencias de la carrera</b>	<ul style="list-style-type: none"> <li>• Demuestra dominio de conocimientos y habilidades para la solución de problemas mediante la aplicación y creación de algoritmos, modelos matemáticos y métodos de Ciencias de la Computación.</li> </ul>	
<b>Contenidos a desarrollarse</b>	Unidad 7: Principios SOLID <ul style="list-style-type: none"> <li>• Interface segregation</li> <li>• Dependency inversion</li> </ul>	
<b>Resultados de aprendizaje</b>	<ul style="list-style-type: none"> <li>• Explica e implementa relaciones de clases en el ámbito de Programación Orientado a Objetos.</li> <li>• Usa subclases para diseñar una jerarquía de clases.</li> <li>• Aplica conceptos de UML en el desarrollo de soluciones computacionales.</li> <li>• Desarrolla aplicaciones usando librerías para la vinculación con gestores de base de datos.</li> </ul>	
<b>Actividades del componente: Aprendizaje en contacto con el docente</b>	<b>Actividades de aprendizaje asistido por el profesor:</b> <ul style="list-style-type: none"> <li>• Clase magistral de contenidos</li> <li>• Resolución de inquietudes</li> </ul> <b>Actividades de aprendizaje colaborativo:</b> <ul style="list-style-type: none"> <li>• Resolución de problemas</li> <li>• Tutorías</li> </ul>	
<b>Horas del componente: Aprendizaje en contacto con el docente</b>	3 horas	
<b>Actividades del componente: Aprendizaje práctico - experimental</b>	<ul style="list-style-type: none"> <li>• Talleres individuales y/o talleres grupales</li> </ul>	
<b>Horas del componente: Aprendizaje práctico - experimental</b>	2 horas	
<b>Actividades del componente: Aprendizaje autónomo</b>	<ul style="list-style-type: none"> <li>• Revisión de contenidos de la unidad 4, 5, 6 y 7; como preparación del evaluación del final del segundo bimestre.</li> </ul>	<ul style="list-style-type: none"> <li>• 4 horas</li> </ul>
<b>Horas del componente: Aprendizaje autónomo</b>	4 horas	

<b>Semana 8</b>		
<b>Competencias de la carrera</b>	<ul style="list-style-type: none"> <li>Demuestra dominio de conocimientos y habilidades para la solución de problemas mediante la aplicación y creación de algoritmos, modelos matemáticos y métodos de Ciencias de la Computación.</li> </ul>	
<b>Contenidos a desarrollarse</b>	<b>Evaluación del segundo bimestre</b>	
<b>Resultados de aprendizaje</b>	<ul style="list-style-type: none"> <li>Explica e implementa relaciones de clases en el ámbito de Programación Orientado a Objetos.</li> <li>Usa subclases para diseñar una jerarquía de clases.</li> <li>Aplica conceptos de UML en el desarrollo de soluciones computacionales.</li> <li>Desarrolla aplicaciones usando librerías para la vinculación con gestores de base de datos</li> </ul>	
<b>Actividades del componente: Aprendizaje en contacto con el docente</b>	<b>Actividades de aprendizaje asistido por el profesor:</b> <ul style="list-style-type: none"> <li>Explicación y resolución de inquietudes durante evaluación bimestral</li> </ul>	
<b>Horas del componente: Aprendizaje en contacto con el docente</b>	3 horas	
<b>Actividades del componente: Aprendizaje práctico - experimental</b>	<ul style="list-style-type: none"> <li>Resolución de ejercicios de recuperación</li> </ul>	
<b>Horas del componente: Aprendizaje práctico - experimental</b>	2 horas	
<b>Actividades del componente: Aprendizaje autónomo</b>	<ul style="list-style-type: none"> <li>Preparación para la evaluación de recuperación</li> <li>Proyecto Bimestral – entrega final</li> </ul>	<ul style="list-style-type: none"> <li>4 horas</li> </ul>
<b>Horas del componente: Aprendizaje autónomo</b>	4 horas	

**Horas de trabajo:** (Totales del bimestre)

**Componente de aprendizaje en contacto con el docente:** 24 horas

**Componente de aprendizaje práctico - experimental:** 16 horas

**Componente de aprendizaje autónomo:** 32 horas

**Fechas importantes:**

- Semana 9: inicio de actividades
- Semana 13: evaluación parcial del segundo bimestre
- Semana 16: evaluación final del segundo bimestre

## Evaluación del segundo bimestre:

Componente	%	Actividad	Instrumento de evaluación	Criterios de Evaluación	Tipo de evaluación	Modalidad evaluación	Ponderación
Aprendizaje en contacto con el docente (3.5 puntos)	35%	Evaluación parcial	Prueba teórica práctica	El estudiante en formación puede: <ul style="list-style-type: none"><li>Estructurar y crear programas bajo el paradigma de Programación Orientada a Objetos.</li><li>Implementar en notación UML y lenguaje de alto nivel el concepto de herencia en Programción Orientada a Objetos.</li><li>Implementar en notación UML y lenguaje de alto nivel el concepto de polimorfismo en Programción Orientada a Objetos.</li><li>Realizar soluciones bajo en el paradigma de Programación a Objetos a través del desarrollo basado en pruebas.</li><li>Aplicar los principios SOLID a soluciones computacionales.</li></ul>	Sumativa	Heteroevaluación	10%
		Evaluación bimestral	Prueba teórica práctica		Sumativa	Heteroevaluación	25%
		SUBTOTAL					
Aprendizaje práctico - experimental (3.5 puntos)	35%	Talleres Individuales	Resolución de ejercicios	El estudiante en formación puede: <ul style="list-style-type: none"><li>Estructurar y crear programas bajo el paradigma de Programación Orientada a Objetos.</li><li>Implementar en notación UML y lenguaje de alto nivel el concepto de herencia en Programción Orientada a Objetos.</li><li>Implementar en notación UML y lenguaje de alto nivel el concepto de polimorfismo en Programción Orientada a Objetos.</li><li>Realizar soluciones bajo en el paradigma de Programación a Objetos a través del desarrollo basado en pruebas.</li><li>Aplicar los principios SOLID a soluciones computacionales.</li></ul>	Sumativa	Heteroevaluación	20%
		Talleres Grupales	Resolución de ejercicios		Sumativa	Heteroevaluación	15%

		SUBTOTAL					35%
Aprendizaje autónomo (3 puntos)	30%	Proyecto bimestral	Resolución de problemas	<div>El estudiante en formación puede:</div> <ul style="list-style-type: none"><li>Estructurar y crear programas bajo el paradigma de Programación Orientada a Objetos.</li><li>Implementar en notación UML y lenguaje de alto nivel el concepto de herencia en Programción Orientada a Objetos.</li><li>Implementar en notación UML y lenguaje de alto nivel el concepto de polimorfismo en Programción Orientada a Objetos.</li><li>Realizar soluciones bajo en el paradigma de Programación a Objetos a través del desarrollo basado en pruebas.</li><li>Aplicar los principios SOLID a soluciones computacionales.</li></ul>	Sumativa	Heteroevaluación	15%
		Talleres individuales Extraclase	Resolución de ejercicios		Sumativa	Heteroevaluación	10%
		Consulta de temáticas	Resúmenes / Investigación		Sumativa	Heteroevaluación	5%
		SUBTOTAL					30 %

TOTAL	10 puntos
-------	-----------

## **E. Evaluación de recuperación**

El estudiante que obtenga una calificación menor a 7 puntos en la nota total final podrá presentarse a la evaluación de recuperación. La ponderación de esta calificación será igual al 35% de la nota (3.5 puntos). Esta calificación será sumada a lo acumulado por el estudiante en los componentes de “Aprendizaje práctico experimental” y “Aprendizaje autónomo”.

## **F. Recursos a utilizar en el desarrollo de la asignatura**

### **Bibliografía básica:**

1. López Román, Leobardo. (2013) *Metodología de la programación orientada a objetos*. México

El texto de Metodología de la programación orientada a objetos cumple con los requerimientos adecuados para ayudar en el aprendizaje de conocimientos relacionados con el paradigma de programación orientado a objetos a través de conceptos y ejemplos haciendo uso de miniespecificaciones o pseudocódigo y diagramas de clases UML (Unified Modeling Language).

**¿El texto está disponible en la biblioteca general física de la UTPL?**

Si

### **Recursos complementarios:**

1. Deitel, H. M., & Deitel, P. J. (2012). *Como programar en Java*. Pearson educación.

La obra permite al lector estudiar de forma teórica y práctica, basándose en una excelente metodología pedagógica, los conceptos, sintaxis y estructura del paradigma de programación orientado a objetos en lenguaje de alto nivel como Java; además, el número elevado de ejercicios permite reforzar conceptos.

**¿El texto está disponible en la biblioteca general física de la UTPL?**

Si

2. Phillips, D. (2015). *Python 3 object-oriented programming*. Packt Publishing Ltd.

La obra ofrece una visión general de los conceptos del lenguaje de programación Python bajo el paradigma de Programación Orientada a Objetos. Indica a los lectores los pasos iniciales para la instalación y puesta en marcha de ejercicios de programación desde la creación de clases y objetos; finalmente se aborda el concepto de herencia y polimorfismo bajo las directrices del lenguaje.

**¿El texto está disponible en la biblioteca general física de la UTPL?**

Si

**Recursos Educativos Abiertos:**

- VIDEO: Escuela de Informáticos.(2016).Programación Orientada a Objetos (POO) - Objetos y Clases. [video]. Recuperado de <https://www.youtube.com/watch?v=ykpT5P7171M>
- VIDEO: ForrestKnight. (2016). How To Create a UML Diagram Using DIA Diagram Editor. [video]. Recuperado de <https://www.youtube.com/watch?v=f-leQbc2o5k>
- LECTURA: Cachero, C., & Ponce de León Amador, P. J. (2011). Tema 2. Fundamentos de la Programación Orientada a Objetos (2010-2011). Programación Orientada a Objetos. Recuperado de <https://rua.ua.es/dspace/handle/10045/15994>
- LECTURA (2011). HERENCIA. <https://rua.ua.es/dspace/bitstream/10045/15995/1/POO-3-Herencia-10-11.pdf>
- LECTURA: (2014) Introduction to Object-Oriented Programming. <http://archive.oreilly.com/oreilyschool/courses/java/>
- LECTURA: Object Oriented Modelling. <http://www.cs.toronto.edu/~sme/CSC340F/slides/11-objects.pdf>
- LECTURA (2019). Serialization in Object-Oriented Programming Languages. [https://www.researchgate.net/publication/335670464\\_Serialization\\_in\\_Object-Oriented\\_Programming\\_Languages](https://www.researchgate.net/publication/335670464_Serialization_in_Object-Oriented_Programming_Languages)
- LECTURA (2017). Object-Oriented Programming in Computer Science. [https://www.researchgate.net/publication/317957956\\_Object-Oriented\\_Programming\\_in\\_Computer\\_Science](https://www.researchgate.net/publication/317957956_Object-Oriented_Programming_in_Computer_Science).

- LECTURA García-Peñalvo, F. J., & Pardo Aguilar, C. Diagramas de Clase en UML 1.1.  
<https://gredos.usal.es/handle/10366/121969>
- LECTURA (2012). Diagramas entidad-relación y de clases de UML. <http://www.cyta.com.ar/ta1101/v11n1a1.htm>
- LECTURA (2010). Diseño Agil con TDD  
<http://ekembi.com/img/capturas/tdd/libro/disenioAgilConTDD.pdf>
- LECTURA (2012). An Improved Java Programming Learning System Using Test-Driven Development Method.  
[http://www.iaeng.org/publication/IMECS2012/IMECS2012\\_pp\\_597-602.pdf](http://www.iaeng.org/publication/IMECS2012/IMECS2012_pp_597-602.pdf)
- LECTURA (2011). SOLID Design for Embedded C.  
[http://www.renaissancesoftware.net/files/articles/ESC-204Paper\\_Grenning-v1r0.pdf](http://www.renaissancesoftware.net/files/articles/ESC-204Paper_Grenning-v1r0.pdf)

### Enlaces Web

- <https://docs.python.org/3/>
- <https://www.oracle.com/technetwork/es/java/javase/documentation/index.html>
- <https://git-scm.com/>
- <https://netbeans.org/>
- <https://github.com/>
- <https://classroom.github.com/>

### Otros recursos:

**NOTA:** Durante todo el bimestre el docente deberá utilizar un portafolio docente digital donde respalde todo el material utilizado para el desarrollo de la asignatura, sean diapositiva, evaluaciones, recursos, etc.

	Nombre	Firma
Elaborado por:	René Rolando Elizalde Solano	
Fecha de elaboración:		
Revisado por:	Pedro Daniel Irene Robalino	
Aprobado por:	(Responsable de sección departamental)	



	Consejo de Departamento, según acta _____ de fecha _____
--	--