

**UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA
VICERRECTORADO ACADÉMICO**

FACULTAD:

INGENIERÍAS Y ARQUITECTURA

CARRERA:

COMPUTACION

**PLAN DOCENTE DE LA
ASIGNATURA:**

INTRODUCCION A LA PROGRAMACION

NOMBRE DEL DOCENTE:

RENE ROLANDO ELIZALDE SOLANO

MODALIDAD DE ESTUDIO:

PRESENCIAL

PERÍODO ACADÉMICO ORDINARIO/SEMESTRE:

OCT/2025 - FEB/2026

A. DATOS BÁSICOS DE LA ASIGNATURA																
Nombre de la asignatura	INTRODUCCION A LA PROGRAMACION															
Código de la asignatura	DSOF_1067															
Campo de formación:	<table><tr><td>Fundamentos Teóricos</td><td>x</td></tr><tr><td>Práxis Profesional</td><td></td></tr><tr><td>Epistemología y Metodología de la Investigación</td><td></td></tr><tr><td>Integración de saberes, contextos y cultura</td><td></td></tr><tr><td>Comunicación y Lenguajes</td><td></td></tr><tr><td>Itinerario</td><td></td></tr></table>				Fundamentos Teóricos	x	Práxis Profesional		Epistemología y Metodología de la Investigación		Integración de saberes, contextos y cultura		Comunicación y Lenguajes		Itinerario	
Fundamentos Teóricos	x															
Práxis Profesional																
Epistemología y Metodología de la Investigación																
Integración de saberes, contextos y cultura																
Comunicación y Lenguajes																
Itinerario																
Unidad de organización curricular:	<table><tr><td>Básica</td><td>Profesional</td><td>Integración Curricular</td></tr><tr><td>x</td><td></td><td></td></tr></table>				Básica	Profesional	Integración Curricular	x								
Básica	Profesional	Integración Curricular														
x																
Número de créditos/horas:	Créditos:	3	Horas:	144												
Carrera:	COMPUTACION															
Facultad:	INGENIERÍAS Y ARQUITECTURA															
Período académico ordinario /Nivel:	PRIMERO															
Período académico ordinario /Semestre:	OCT/2025 - FEB/2026															

B. DESCRIPCIÓN DE LA ASIGNATURA	
Prerequisitos:	La asignatura sugiere que el estudiante tenga como pre-requisitos una fundamentación matemática básica para ingeniería. Además, como co-requisito se incluye la asignatura de Fundamentos Computacionales.

Contribución de la asignatura en el perfil de egreso de la carrera

Resultados de aprendizaje de la asignatura	Contribución (inicial, medio, final)	Resultados de aprendizaje del perfil de egreso
Analiza y explica el comportamiento de programas que involucran estructuras fundamentales de programación como variables, expresiones, asignaciones, archivos, estructuras de decisión y control, arreglos, funciones, paso de parámetros, y recursividad	Inicial	Analizar problemas computacionales y plantear soluciones mediante métodos matemáticos
Diseña, implementa, prueba, y depura un programa que usa estructuras fundamentales de programación	Inicial	Analizar problemas computacionales y plantear soluciones mediante métodos matemáticos
Identifica y previene errores de codificación comunes a nivel lógico y de sintaxis, a través del uso de excepciones	Inicial	Analizar problemas computacionales y plantear soluciones mediante métodos matemáticos
Aplica estrategias para la prueba y depuración de programas	Inicial	Analizar problemas computacionales y plantear soluciones mediante métodos matemáticos
Construye, ejecuta y depura programas usando un IDE moderno	Inicial	Analizar problemas computacionales y plantear soluciones mediante métodos matemáticos
Aplica estándares de documentación y estilo de programación que contribuyan a la legibilidad y mantenimiento de software	Inicial	Analizar problemas computacionales y plantear soluciones mediante métodos matemáticos
Extrae y manipula información de archivos de datos	Inicial	Analizar problemas computacionales y plantear soluciones mediante métodos matemáticos

Organización del aprendizaje

Organización del aprendizaje	
Componente	Número de horas
Aprendizaje en contacto con el docente	48
Aprendizaje práctico - experimental	32
Aprendizaje autónomo	64
Total	144

Horario de clases:

Docente			Paralelo Día		Aula	Horario
RENE SOLANO	ROLANDO ELIZALDE		A	JUEVES	AULA 0511 LAB. DE ELECTRÓNICA	10:00 AM-11:59 AM0 (PRAC)
RENE SOLANO	ROLANDO ELIZALDE		A	MIERCOLES	AULA 0511 LAB. DE ELECTRÓNICA	11:00 AM-12:59 PM0 (CLAS)

RENE ROLANDO ELIZALDE SOLANO	A	JUEVES	AULA 0511 LAB. DE ELECTRÓNICA	12:00 PM-12:59 PM0 (TUTO)
ANGEL EDUARDO ENCALADA ENCALADA	B	MIERCOLES	AULA H COMPUTO	03:00 PM-04:59 PM0 (CLAS)
ANGEL EDUARDO ENCALADA ENCALADA	B	JUEVES	AULA H COMPUTO	06:00 PM-07:59 PM0 (PRAC)
ANGEL EDUARDO ENCALADA ENCALADA	B	JUEVES	AULA H COMPUTO	08:00 PM-08:59 PM0 (TUTO)
PEDRO DANIEL IRENE ROBALINO	C	JUEVES	AULA E COMPUTO	10:00 AM-11:59 AM0 (PRAC)
PEDRO DANIEL IRENE ROBALINO	C	MIERCOLES	AULA E COMPUTO	11:00 AM-12:59 PM0 (CLAS)
PEDRO DANIEL IRENE ROBALINO	C	JUEVES	AULA E COMPUTO	12:00 PM-12:59 PM0 (TUTO)
PEDRO DANIEL IRENE ROBALINO	D	JUEVES	AULA E COMPUTO	06:00 PM-07:59 PM0 (PRAC)
PEDRO DANIEL IRENE ROBALINO	D	MIERCOLES	AULA E COMPUTO	03:00 PM-04:59 PM0 (CLAS)
PEDRO DANIEL IRENE ROBALINO	D	JUEVES	AULA E COMPUTO	08:00 PM-08:59 PM0 (TUTO)
ANGEL EDUARDO ENCALADA ENCALADA	E	JUEVES	AULA C COMPUTO	10:00 AM-11:59 AM0 (PRAC)
ANGEL EDUARDO ENCALADA ENCALADA	E	MIERCOLES	AULA C COMPUTO	11:00 AM-12:59 PM0 (CLAS)
ANGEL EDUARDO ENCALADA ENCALADA	E	JUEVES	AULA C COMPUTO	12:00 PM-12:59 PM0 (TUTO)

Adaptaciones curriculares:

Para garantizar una educación de calidad acorde a las características del modelo educativo de la Universidad Técnica Particular de Loja, al principio de igualdad de oportunidades y a las necesidades educativas especiales asociadas o no a la discapacidad, se desarrollan adaptaciones curriculares no significativas o de grado dos, mismas que siguen una trayectoria de menor a mayor significación; considerando el aspecto metodológico, actividades de aprendizaje y el estilo individual de aprendizaje en cuanto a las estrategias a desarrollar. Estas adaptaciones se realizan en función de la identificación de las necesidades educativas en las primeras semanas de trabajo académico, con la finalidad de dar respuesta a la dificultad de aprendizaje y apoyar al desarrollo de las competencias del estudiante.

C. COMPETENCIAS DEL PERFIL PROFESIONAL A LAS QUE APORTA LA ASIGNATURA

Definir, diseñar, implementar, mantener y evaluar soluciones computacionales basadas en atributos de calidad internacionales que permitan la toma de decisiones sobre grandes volúmenes de datos, monitoreando e interpretando cambios para generar nuevas ideas que fomenten la investigación, el desarrollo e innovación tecnológica, en colaboración con equipos multidisciplinarios integrando valores éticos y normas sociales adecuadas

D. PLANIFICACIÓN GENERAL DE LA ASIGNATURA

Primer Bimestre

Semana 1		
Resultados de aprendizaje de la asignatura	<ul style="list-style-type: none">Analiza y explica el comportamiento de programas que involucran estructuras fundamentales de programación como variables, expresiones, asignaciones, archivos, estructuras de decisión y control, arreglos, funciones, paso de parámetros, y recursividad	
Contenidos a desarrollarse	Unidad 1: Introducción a Programación. <ul style="list-style-type: none">IntroducciónImportancia de la programaciónLenguajes de ProgramaciónMetodologías de ProgramaciónEntornos de Programación	
Actividades del componente: Aprendizaje en contacto con el docente	<ul style="list-style-type: none">Clase magistral de contenidos Exposición y explicación de contenidosResolución de inquietudes Respuesta a preguntas por parte de los estudiantes, sobre las temáticas estudiadas.Resolución de problemas Planteamiento y guía en la solución de problemasTutorías TutoríasEvaluación diagnóstica Evaluación diagnóstica de competencias previas.	
Horas del componente: Aprendizaje en contacto con el docente	3.0	
Actividades del componente: Aprendizaje práctico - experimental	<ul style="list-style-type: none">Taller Individual 1 (Laboratorio) Instalación, configuración del entorno de trabajo.	
Horas del componente: Aprendizaje práctico - experimental	2.0	
Actividades del	<div>[Lecturas] Temas de la Unidad 2 1. Recurso [10] Pro Git 2. Recurso [9]: Del cloud computing al big data</div>	<div>2.0</div>

componente: Aprendizaje autónomo	<div>Consulta 01</div> <div>[Consulta] Investigar sobre Entornos de virtualización para pruebas y desarrollo – Docker.</div>	2.0
Horas del componente: Aprendizaje autónomo	4.0	

Semana 2		
Resultados de la aprendizaje de la asignatura	<ul style="list-style-type: none"> • Aplica estándares de documentación y estilo de programación que contribuyan a la legibilidad y mantenimiento de software 	
Contenidos a desarrollarse	Unidad 2: Entornos modernos de desarrollo. <ul style="list-style-type: none"> • Gestión de código fuente a través de software de control de versiones. • Entornos de virtualización para pruebas y desarrollo. • Introducción Computación en la nube (Cloud Computing). 	
Actividades del componente: Aprendizaje en contacto con el docente	<ul style="list-style-type: none"> • Clase magistral de contenidos Exposición y explicación de contenidos • Resolución de inquietudes Respuesta a preguntas por parte de los estudiantes, sobre las temáticas estudiadas. • Resolución de problemas Planteamiento y guía en la solución de problemas • Tutorías Tutorías 	
Horas del componente: Aprendizaje en contacto con el docente	3.0	
Actividades del componente: Aprendizaje práctico - experimental	<ul style="list-style-type: none"> • Taller 2 Individual (Laboratorio) Configuración de uso de Git y GitHub 	
Horas del componente: Aprendizaje práctico - experimental	2.0	

Actividades componente: Aprendizaje autónomo	del	<p>Lectura</p> <p>Lectura comprensiva de los temas de la Unidad 3 en los siguientes recursos:</p> <p>Bibliografía básica [1] Unidades: Introducción a las aplicaciones en Java: entrada /salida y operadores</p>	1.0
		<p>Lectura</p> <p>Lectura comprensiva de los temas de la Unidad 3 en los siguientes recursos:</p> <p>Recurso [11]: Java y sus características; Variables, expresiones y variables</p>	1.0
		<p>Lectura</p> <p>Lectura comprensiva de los temas de la Unidad 3 en los siguientes recursos:</p> <p>Bibliografía Complementaria [17] Unidades:</p> <p>Elementos para solucionar problemas en pseudocódigo: La secuenciación.</p>	1.0
		<p>Aprendizaje Basado en Proyectos semana 2</p> <p>Aprendizaje Basado en Proyectos</p> <p>Actividad: Configurar entorno y repositorio; documentar objetivos y entradas /salidas</p>	1.0
Horas componente: Aprendizaje autónomo	del	4.0	

Semana 3			
Resultados aprendizaje asignatura	de de la	<ul style="list-style-type: none"> Analiza y explica el comportamiento de programas que involucren estructuras fundamentales de programación como variables, expresiones, asignaciones, archivos, estructuras de decisión y control, arreglos, funciones, paso de parámetros, y recursividad Construye, ejecuta y depura programas usando un IDE moderno Aplica estándares de documentación y estilo de programación que contribuyan a la legibilidad y mantenimiento de software 	
Contenidos desarrollarse	a	<p>Unidad 3: Sintaxis y Semántica de lenguajes de programación.</p> <ul style="list-style-type: none"> Elementos básicos de la programación. Manejo de tipos de datos en lenguajes de alto nivel. Manejo de operadores y expresiones en lenguaje de alto nivel. 	
Actividades	del	<ul style="list-style-type: none"> Clase magistral de contenidos Exposición y explicación de contenidos Resolución de inquietudes Respuesta a preguntas por parte de los estudiantes, sobre las temáticas estudiadas. 	

componente: Aprendizaje en contacto con el docente	<ul style="list-style-type: none"> Resolución de problemas <p>Planteamiento y guía en la solución de problemas</p> <ul style="list-style-type: none"> Tutorías <p>Tutorías</p>				
Horas del componente: Aprendizaje en contacto con el docente	3.0				
Actividades del componente: Aprendizaje práctico - experimental	<ul style="list-style-type: none"> Taller 3 Grupal (Laboratorio) <p>Manejo de tipos de datos, operadores y expresiones en lenguajes de alto nivel.</p>				
Horas del componente: Aprendizaje práctico - experimental	2.0				
Actividades del componente: Aprendizaje autónomo	<table border="1"> <tr> <td> Lectura Lectura comprensiva de los temas de la Unidad 3 en los siguientes recursos: Bibliografía básica [1] Unidades: Introducción a las aplicaciones en Java: entrada /salida y operadores </td><td>2.0</td></tr> <tr> <td> Lectura Lectura comprensiva de los temas de la Unidad 3 en los siguientes recursos: Bibliografía Básica[2]: Estructura de una aplicación en Java Bibliografía [17] Unidades: Elementos para solucionar problemas en pseudocódigo. La secuenciación. </td><td>2.0</td></tr> </table>	Lectura Lectura comprensiva de los temas de la Unidad 3 en los siguientes recursos: Bibliografía básica [1] Unidades: Introducción a las aplicaciones en Java: entrada /salida y operadores	2.0	Lectura Lectura comprensiva de los temas de la Unidad 3 en los siguientes recursos: Bibliografía Básica[2]: Estructura de una aplicación en Java Bibliografía [17] Unidades: Elementos para solucionar problemas en pseudocódigo. La secuenciación.	2.0
Lectura Lectura comprensiva de los temas de la Unidad 3 en los siguientes recursos: Bibliografía básica [1] Unidades: Introducción a las aplicaciones en Java: entrada /salida y operadores	2.0				
Lectura Lectura comprensiva de los temas de la Unidad 3 en los siguientes recursos: Bibliografía Básica[2]: Estructura de una aplicación en Java Bibliografía [17] Unidades: Elementos para solucionar problemas en pseudocódigo. La secuenciación.	2.0				
Horas del componente: Aprendizaje autónomo	4.0				

Semana 4

Resultados de aprendizaje de la asignatura	<ul style="list-style-type: none"> Analiza y explica el comportamiento de programas que involucran estructuras fundamentales de programación como variables, expresiones, asignaciones, archivos, estructuras de decisión y control, arreglos, funciones, paso de parámetros, y recursividad Construye, ejecuta y depura programas usando un IDE moderno Aplica estándares de documentación y estilo de programación que contribuyan a la legibilidad y mantenimiento de software
	Unidad 3: Sintaxis y Semántica de lenguajes de programación

Contenidos a desarrollarse	<ul style="list-style-type: none"> • Creación de programas en lenguajes de alto nivel • Depuración de programas 	
Actividades del componente: Aprendizaje en contacto con el docente	<ul style="list-style-type: none"> • Clase magistral de contenidos Exposición y explicación de contenidos • Resolución de inquietudes Respuesta a preguntas por parte de los estudiantes, sobre las temáticas estudiadas. • Resolución de problemas Planteamiento y guía en la solución de problemas • Tutorías Tutorías 	
Horas del componente: Aprendizaje en contacto con el docente	3.0	
Actividades del componente: Aprendizaje práctico - experimental	<ul style="list-style-type: none"> • Taller 4 Individual (Laboratorio): Construcción de programas básicos 	
Horas del componente: Aprendizaje práctico - experimental	2.0	
Actividades del componente: Aprendizaje autónomo	Lectura Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos: Bibliografía básica [1] Unidades: Instrucciones de control: parte 1	0.5
	Lectura Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos: Bibliografía básica [2] Unidades: Estructuras de control Bibliografía [17] Unidades: La selección	0.5
	Lectura Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos: Recurso [11]: Condicionales	1.0
	Lectura	

	<p>Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:</p> <p>Recurso [12]. Elementos de programación estructurada</p>	1.0
	<p>Aprendizaje Basado en Proyectos semana 4</p> <p>Aprendizaje Basado en Proyectos</p> <p>Actividad: Implementar versión inicial de entrada/salida y validaciones simples según pseudocódigo</p>	1.0
Horas del componente: Aprendizaje autónomo	4.0	

Semana 5		
Resultados de aprendizaje de la asignatura	<ul style="list-style-type: none"> Analiza y explica el comportamiento de programas que involucren estructuras fundamentales de programación como variables, expresiones, asignaciones, archivos, estructuras de decisión y control, arreglos, funciones, paso de parámetros, y recursividad Construye, ejecuta y depura programas usando un IDE moderno Aplica estándares de documentación y estilo de programación que contribuyan a la legibilidad y mantenimiento de software 	
Contenidos a desarrollar	<p>Unidad 4: Estructuras de control</p> <ul style="list-style-type: none"> Estructuras de selección simple - if Estructuras de selección compuestas - if/else 	
Actividades del componente: Aprendizaje en contacto con el docente	<ul style="list-style-type: none"> Clase magistral de contenidos Exposición y explicación de contenidos Resolución de inquietudes Respuesta a preguntas por parte de los estudiantes, sobre las temáticas estudiadas. Resolución de problemas Planteamiento y guía en la solución de problemas Tutorías Tutorías Evaluación parcial del primer bimestre Resolución de las preguntas y problemas de la evaluación parcial del primer bimestre por parte de los estudiantes. 	
Horas del componente: Aprendizaje en contacto con el docente	3.00	

Actividades del componente: Aprendizaje práctico - experimental	<ul style="list-style-type: none"> Taller 5 Grupal (Laboratorio) <p>Construcción de programas usando estructuras de selección simple y compuesta</p>								
Horas del componente: Aprendizaje práctico - experimental	2.0								
Actividades del componente: Aprendizaje autónomo	<table> <tr> <td> <p>Lectura</p> <p>Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:</p> <p>Bibliografía básica [1] Unidades:</p> <p>Instrucciones de control: parte 1</p> </td><td>1.0</td></tr> <tr> <td> <p>Lectura</p> <p>Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:</p> <p>Bibliografía básica [2] Unidades: Estructuras de Control</p> <p>Bibliografía [17] Unidades: La selección</p> </td><td>1.0</td></tr> <tr> <td> <p>Lectura</p> <p>Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:</p> <p>Recurso [11]: Condicionales</p> </td><td>1.0</td></tr> <tr> <td> <p>Lectura</p> <p>Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:</p> <p>Recurso [12]. Elementos de programación estructurada</p> </td><td>1.0</td></tr> </table>	<p>Lectura</p> <p>Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:</p> <p>Bibliografía básica [1] Unidades:</p> <p>Instrucciones de control: parte 1</p>	1.0	<p>Lectura</p> <p>Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:</p> <p>Bibliografía básica [2] Unidades: Estructuras de Control</p> <p>Bibliografía [17] Unidades: La selección</p>	1.0	<p>Lectura</p> <p>Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:</p> <p>Recurso [11]: Condicionales</p>	1.0	<p>Lectura</p> <p>Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:</p> <p>Recurso [12]. Elementos de programación estructurada</p>	1.0
<p>Lectura</p> <p>Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:</p> <p>Bibliografía básica [1] Unidades:</p> <p>Instrucciones de control: parte 1</p>	1.0								
<p>Lectura</p> <p>Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:</p> <p>Bibliografía básica [2] Unidades: Estructuras de Control</p> <p>Bibliografía [17] Unidades: La selección</p>	1.0								
<p>Lectura</p> <p>Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:</p> <p>Recurso [11]: Condicionales</p>	1.0								
<p>Lectura</p> <p>Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:</p> <p>Recurso [12]. Elementos de programación estructurada</p>	1.0								
Horas del componente: Aprendizaje autónomo	4.0								

Semana 6

Resultados de aprendizaje de la asignatura	<ul style="list-style-type: none"> Analiza y explica el comportamiento de programas que involucren estructuras fundamentales de programación como variables, expresiones, asignaciones, archivos, estructuras de decisión y control, arreglos, funciones, paso de parámetros, y recursividad Construye, ejecuta y depura programas usando un IDE moderno Aplica estándares de documentación y estilo de programación que contribuyan a la legibilidad y mantenimiento de software
---	--

Contenidos a desarrollarse	Unidad 4: Estructuras de control. <ul style="list-style-type: none"> Estructuras de selección anidadas. Estructura de control múltiple (switch). 		
Actividades del componente: Aprendizaje en contacto con el docente	<ul style="list-style-type: none"> Clase magistral de contenidos Exposición y explicación de contenidos Resolución de inquietudes Respuesta a preguntas por parte de los estudiantes, sobre las temáticas estudiadas. Resolución de problemas Planteamiento y guía en la solución de problemas Tutorías Tutorías 		
Horas del componente: Aprendizaje en contacto con el docente	3.0		
Actividades del componente: Aprendizaje práctico - experimental	<ul style="list-style-type: none"> Práctica - Generación de programas usando estructuras de selección simples, compuestas, anidadas y switch. Generación de programas usando estructuras de selección simples, compuestas, anidadas y switch. Taller 6 (Laboratorio) Construcción de programas usando estructuras de selección anidadas y switch. 		
Horas del componente: Aprendizaje práctico - experimental	2.0		
	Lectura Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos: Bibliografía básica [1] Unidades: Instrucciones de control: parte 1 Instrucciones de control: parte 2		1.0
	Lectura Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos: Bibliografía básica [2] Unidades: Estructuras de control Bibliografía [17] Unidades: La repetición		1.0

Actividades del componente: Aprendizaje autónomo		
	<p>Lectura</p> <p>Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:</p> <p>Recurso [11]: Iteración</p>	0.5
	<p>Lectura</p> <p>Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:</p> <p>Recurso [12]. Elementos de programación estructurada</p>	0.5
	<p>Aprendizaje Basado en Proyectos semana 6</p> <p>Aprendizaje Basado en Proyectos</p> <p>Actividad: Desarrollar la primera versión funcional del programa (variables y reglas básicas)</p>	1.0
Horas del componente: Aprendizaje autónomo	4.0	

Semana 7		
Resultados de aprendizaje de la asignatura	<ul style="list-style-type: none"> Analiza y explica el comportamiento de programas que involucran estructuras fundamentales de programación como variables, expresiones, asignaciones, archivos, estructuras de decisión y control, arreglos, funciones, paso de parámetros, y recursividad Construye, ejecuta y depura programas usando un IDE moderno Aplica estándares de documentación y estilo de programación que contribuyan a la legibilidad y mantenimiento de software 	
Contenidos a desarrollarse	<p>Unidad 4: Estructuras de control.</p> <ul style="list-style-type: none"> Estructuras repetitivas while. Estructuras repetitivas do-while. 	
	<ul style="list-style-type: none"> Resolución de problemas <p>Planteamiento y guía en la solución de problemas</p> <ul style="list-style-type: none"> Tutorías 	

Actividades del componente: Aprendizaje en contacto con el docente	<p>Tutorías</p> <ul style="list-style-type: none"> • Clase magistral de contenidos <p>Exposición y explicación de contenidos</p> <ul style="list-style-type: none"> • Resolución de inquietudes <p>Respuesta a preguntas por parte de los estudiantes, sobre las temáticas estudiadas.</p>		
Horas del componente: Aprendizaje en contacto con el docente	3.0		
Actividades del componente: Aprendizaje práctico - experimental	<ul style="list-style-type: none"> • Taller 7 Grupal (Laboratorio) <p>Construcción de programas usando estructuras de control repetitivas (do / do-while)</p>		
Horas del componente: Aprendizaje práctico - experimental	2.0		
Actividades del componente: Aprendizaje autónomo	<table border="1"> <tr> <td> <p>Lectura</p> <p>Revisión de contenidos de las unidades como preparación de la evaluación final del primer bimestre.</p> </td> <td>4.0</td> </tr> </table>	<p>Lectura</p> <p>Revisión de contenidos de las unidades como preparación de la evaluación final del primer bimestre.</p>	4.0
<p>Lectura</p> <p>Revisión de contenidos de las unidades como preparación de la evaluación final del primer bimestre.</p>	4.0		
Horas del componente: Aprendizaje autónomo	4.0		

Semana 8	
Resultados de aprendizaje de la asignatura	<ul style="list-style-type: none"> • Analiza y explica el comportamiento de programas que involucren estructuras fundamentales de programación como variables, expresiones, asignaciones, archivos, estructuras de decisión y control, arreglos, funciones, paso de parámetros, y recursividad • Construye, ejecuta y depura programas usando un IDE moderno • Aplica estándares de documentación y estilo de programación que contribuyan a la legibilidad y mantenimiento de software
Contenidos a desarrollarse	<p>Repaso Primer Bimestre</p> <p>Evaluación del primer bimestre</p>
Actividades del componente: Aprendizaje en contacto con el docente	<ul style="list-style-type: none"> • Evaluación Bimestral <p>Explicación y resolución de inquietudes durante evaluación bimestral</p> <ul style="list-style-type: none"> • Tutorías <p>Tutorías</p>

Horas del componente: Aprendizaje en contacto con el docente	3.0				
Actividades del componente: Aprendizaje práctico - experimental	<ul style="list-style-type: none">Resolución guiada de la evaluación bimestral <p>Explicación de las preguntas y problemas de la evaluación bimestral</p>				
Horas del componente: Aprendizaje práctico - experimental	2.0				
Actividades del componente: Aprendizaje autónomo	<table><tr><td>Aprendizaje Basado en Proyectos semana 8</td><td rowspan="3">2.0</td></tr><tr><td>Aprendizaje Basado en Proyectos</td></tr><tr><td>Actividad: Exposición parcial: demo del programa básico y coherencia con diseño de Fundamentos de Programación</td></tr></table>	Aprendizaje Basado en Proyectos semana 8	2.0	Aprendizaje Basado en Proyectos	Actividad: Exposición parcial: demo del programa básico y coherencia con diseño de Fundamentos de Programación
	Aprendizaje Basado en Proyectos semana 8	2.0			
	Aprendizaje Basado en Proyectos				
	Actividad: Exposición parcial: demo del programa básico y coherencia con diseño de Fundamentos de Programación				
<table><tr><td>Lectura</td><td rowspan="3">1.0</td></tr><tr><td>Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:</td></tr><tr><td>Bibliografía básica [1] Unidades: Instrucciones de control: parte 2</td></tr></table>	Lectura	1.0	Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:	Bibliografía básica [1] Unidades: Instrucciones de control: parte 2	
Lectura	1.0				
Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:					
Bibliografía básica [1] Unidades: Instrucciones de control: parte 2					
<table><tr><td>Lectura</td><td rowspan="4">0.5</td></tr><tr><td>Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:</td></tr><tr><td>Bibliografía básica [2] Unidades: Estructuras de control</td></tr><tr><td>Bibliografía [17] Unidades: La repetición</td></tr></table>	Lectura	0.5	Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:	Bibliografía básica [2] Unidades: Estructuras de control	Bibliografía [17] Unidades: La repetición
Lectura	0.5				
Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:					
Bibliografía básica [2] Unidades: Estructuras de control					
Bibliografía [17] Unidades: La repetición					
<table><tr><td>Lectura</td><td rowspan="3">0.5</td></tr><tr><td>Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:</td></tr><tr><td>Recurso [11]: Iteración</td></tr></table>	Lectura	0.5	Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:	Recurso [11]: Iteración	
Lectura	0.5				
Lectura comprensiva de los temas de la Unidad 4 en los siguientes recursos:					
Recurso [11]: Iteración					
Horas del componente: Aprendizaje autónomo	4.0				

FECHAS IMPORTANTES Primer Bimestre:

- Clase presencial: Semana 1 - del 06/10/2025 al 12/10/2025
- Proyecto de investigación: Semana 2 - del 13/10/2025 al 19/10/2025
- Proyecto de investigación: Semana 4 - del 27/10/2025 al 02/11/2025
- Prueba parcial: Semana 5 - del 03/11/2025 al 09/11/2025
- Prácticas: Semana 6 - del 10/11/2025 al 16/11/2025
- Proyecto de investigación: Semana 6 - del 10/11/2025 al 16/11/2025
- Examen bimestral: Semana 8 - del 24/11/2025 al 30/11/2025
- Proyecto de investigación: Semana 8 - del 24/11/2025 al 30/11/2025

EVALUACIÓN DEL Primer Bimestre

Componente	Porcentaje	Actividad	Instrumento de evaluación	Tipo de evaluación	Modalidad de evaluación	Ponderación
Aprendizaje en contacto con el docente (3.5 puntos)	35%	Examen bimestral	Cuestionario	Sumativa	Heteroevaluación	25.00 %
		Prueba parcial	Cuestionario	Sumativa	Heteroevaluación	10.00 %
Aprendizaje práctico - experimental (3.5 puntos)	35%	Taller individual	Resolución de ejercicios	Sumativa	Heteroevaluación	20.00 %
		Taller grupal	Resolución de ejercicios	Sumativa	Heteroevaluación	15.00 %
		Prácticas		Formativa		
Aprendizaje autónomo (3.0 puntos)	30%	Búsqueda de información	Resúmenes	Sumativa	Heteroevaluación	5.00 %
		Proyecto de investigación	Resolución de problemas	Sumativa	Heteroevaluación	25.00 %

Segundo Bimestre

Semana 9	
Resultados de aprendizaje de la asignatura	<ul style="list-style-type: none"> Analiza y explica el comportamiento de programas que involucren estructuras fundamentales de programación como variables, expresiones, asignaciones, archivos, estructuras de decisión y control, arreglos, funciones, paso de parámetros, y recursividad Construye, ejecuta y depura programas usando un IDE moderno Aplica estándares de documentación y estilo de programación que contribuyan a la legibilidad y mantenimiento de software
Contenidos a desarrollarse	Unidad 4: Estructuras de control. <ul style="list-style-type: none"> Estructuras repetitivas for.
Actividades del componente: Aprendizaje en contacto con el docente	<ul style="list-style-type: none"> Clase magistral de contenidos Exposición y explicación de contenidos Resolución de inquietudes Respuesta a preguntas por parte de los estudiantes, sobre las temáticas estudiadas. Resolución de problemas Planteamiento y guía en la solución de problemas Tutorías

	Tutorías	
Horas del componente: Aprendizaje en contacto con el docente	3.0	
Actividades del componente: Aprendizaje práctico - experimental	<ul style="list-style-type: none">Taller 8 Individual (Laboratorio) <p>Construcción de programas usando estructuras de control repetitivas (for)</p>	
Horas del componente: Aprendizaje práctico - experimental	2.0	
Actividades del componente: Aprendizaje autónomo	<div>Lectura</div> <div>Lectura comprensiva de los temas de la unidad 5 en los siguientes recursos:</div> <div>Bibliografía básica [1] Unidades: Arreglos y objetos ArrayList</div>	2.0
	<div>Lectura</div> <div>Lectura comprensiva de los temas de la unidad 5 en los siguientes recursos:</div> <div>Bibliografía básica [2] Unidades: Colecciones y Estructuras de Datos</div> <div>Bibliografía [17] Unidades: Arreglos</div>	1.0
	<div>Lectura</div> <div>Lectura comprensiva de los temas de la unidad 5 en los siguientes recursos:</div> <div>Recurso [11]: Arreglos</div>	1.0
Horas del componente: Aprendizaje autónomo	4.0	

Semana 10	

Resultados de aprendizaje de la asignatura	<ul style="list-style-type: none"> • Analiza y explica el comportamiento de programas que involucran estructuras fundamentales de programación como variables, expresiones, asignaciones, archivos, estructuras de decisión y control, arreglos, funciones, paso de parámetros, y recursividad • Diseña, implementa, prueba, y depura un programa que usa estructuras fundamentales de programación • Construye, ejecuta y depura programas usando un IDE moderno • Aplica estándares de documentación y estilo de programación que contribuyan a la legibilidad y mantenimiento de software 		
Contenidos a desarrollarse	Unidad 5: Estructuras de Datos <ul style="list-style-type: none"> • Arreglos Unidimensionales 		
Actividades del componente: Aprendizaje en contacto con el docente	<ul style="list-style-type: none"> • Clase magistral de contenidos Exposición y explicación de contenidos • Resolución de inquietudes Respuesta a preguntas por parte de los estudiantes, sobre las temáticas estudiadas. • Resolución de problemas Planteamiento y guía en la solución de problemas • Tutorías Tutorías 		
Horas del componente: Aprendizaje en contacto con el docente	3.0		
Actividades del componente: Aprendizaje práctico - experimental	<ul style="list-style-type: none"> • Taller 9 Grupal (Laboratorio) Construcción de programas usando arreglos unidimensionales 		
Horas del componente: Aprendizaje práctico - experimental	2.0		
Actividades del componente: Aprendizaje autónomo	Lectura Lectura comprensiva de los temas de la unidad 5 en los siguientes recursos: Bibliografía básica [1] Unidades: Arreglos y objetos ArrayList		1.0
	Lectura Lectura comprensiva de los temas de la unidad 5 en los siguientes recursos: Bibliografía básica [2] Unidades: Colecciones y Estructuras de Datos Bibliografía [17] Unidades: Arreglos		1.0

	<p>Aprendizaje Basado en Proyectos semana 10</p> <p>Aprendizaje Basado en Proyectos</p> <p>Actividad: Refactorizar el código incorporando estructuras de datos (arreglos, listas, etc.)</p>	2.0
Horas del componente: Aprendizaje autónomo	4.0	

Semana 11		
Resultados de aprendizaje de la asignatura	<ul style="list-style-type: none"> Analiza y explica el comportamiento de programas que involucran estructuras fundamentales de programación como variables, expresiones, asignaciones, archivos, estructuras de decisión y control, arreglos, funciones, paso de parámetros, y recursividad Diseña, implementa, prueba, y depura un programa que usa estructuras fundamentales de programación Construye, ejecuta y depura programas usando un IDE moderno Aplica estándares de documentación y estilo de programación que contribuyan a la legibilidad y mantenimiento de software 	
Contenidos a desarrollarse	<p>Unidad 5: Estructuras de Datos</p> <ul style="list-style-type: none"> Arreglos Bidimensionales 	
Actividades del componente: Aprendizaje en contacto con el docente	<ul style="list-style-type: none"> Clase magistral de contenidos <p>Exposición y explicación de contenidos</p> <ul style="list-style-type: none"> Resolución de inquietudes <p>Respuesta a preguntas por parte de los estudiantes, sobre las temáticas estudiadas.</p> <ul style="list-style-type: none"> Resolución de problemas <p>Planteamiento y guía en la solución de problemas</p> <ul style="list-style-type: none"> Tutorías <p>Tutorías</p>	
Horas del componente: Aprendizaje en contacto con el docente	3.0	
Actividades del componente: Aprendizaje práctico - experimental	<ul style="list-style-type: none"> Taller 10 Individual (Laboratorio) <p>Construcción de programas usando arreglos bidimensionales</p>	
Horas del componente: Aprendizaje práctico - experimental	2.0	

Actividades del componente: Aprendizaje autónomo	<p>Lectura</p> <p>Lectura comprensiva de los temas de las unidades 6 y 7 en los siguientes recursos:</p> <p>Bibliografía básica [1] Unidades: Métodos: un análisis más detallado, Recursividad</p>	0.5
	<p>Lectura</p> <p>Lectura comprensiva de los temas de las unidades 6 y 7 en los siguientes recursos:</p> <p>Bibliografía [17] Unidades: Métodos</p>	0.5
	<p>Lectura</p> <p>Lectura comprensiva de los temas de las unidades 6 y 7 en los siguientes recursos:</p> <p>Recurso [11]: Métodos</p>	1.0
	<p>Estudio de caso práctico</p> <p>Implementación en lenguaje de alto nivel de problemática dada (avance 1)</p>	2.0
Horas del componente: Aprendizaje autónomo	4.0	

Semana 12

Resultados de aprendizaje de la asignatura	<ul style="list-style-type: none"> Analiza y explica el comportamiento de programas que involucran estructuras fundamentales de programación como variables, expresiones, asignaciones, archivos, estructuras de decisión y control, arreglos, funciones, paso de parámetros, y recursividad Diseña, implementa, prueba, y depura un programa que usa estructuras fundamentales de programación Construye, ejecuta y depura programas usando un IDE moderno Aplica estándares de documentación y estilo de programación que contribuyan a la legibilidad y mantenimiento de software
Contenidos a desarrollarse	<p>Unidad 6: Módulos y Funciones</p> <ol style="list-style-type: none"> Introducción Declaración de funciones y métodos en lenguajes de alto nivel Ámbito de variables locales y globales Manejo de métodos que no regresen valor. Manejo de métodos que regresen valor. Sobrecarga de métodos
	<ul style="list-style-type: none"> Clase magistral de contenidos <p>Exposición y explicación de contenidos</p>

Actividades del componente: Aprendizaje en contacto con el docente	<ul style="list-style-type: none"> Resolución de inquietudes Respuesta a preguntas por parte de los estudiantes, sobre las temáticas estudiadas. Resolución de problemas Planteamiento y guía en la solución de problemas Tutorías Tutorías 						
Horas del componente: Aprendizaje en contacto con el docente	3.0						
Actividades del componente: Aprendizaje práctico - experimental	<ul style="list-style-type: none"> Taller 11 Individual (Laboratorio) Construcción de programas usando módulos y funciones Práctica - Generación de programas usando usando arreglos unidimensionales /bidimensionales/funciones y procedimientos Generación de programas usando usando arreglos unidimensionales/bidimensionales /funciones y procedimientos 						
Horas del componente: Aprendizaje práctico - experimental	2.0						
Actividades del componente: Aprendizaje autónomo	<table border="1"> <tr> <td> Lectura Lectura comprensiva de los temas de la unidad 8 en los siguientes recursos: Bibliografía básica [1] Unidades: Recursividad </td> <td>2.0</td> </tr> <tr> <td> Aprendizaje Basado en Proyectos semana 12 Aprendizaje Basado en Proyectos Actividad: Modularizar el código en funciones; implementar procesos de comparación o análisis </td> <td>1.0</td> </tr> <tr> <td> Estudio de caso práctico Implementación en lenguaje de alto nivel de problemática dada (avance 2) </td> <td>1.0</td> </tr> </table>	Lectura Lectura comprensiva de los temas de la unidad 8 en los siguientes recursos: Bibliografía básica [1] Unidades: Recursividad	2.0	Aprendizaje Basado en Proyectos semana 12 Aprendizaje Basado en Proyectos Actividad: Modularizar el código en funciones; implementar procesos de comparación o análisis	1.0	Estudio de caso práctico Implementación en lenguaje de alto nivel de problemática dada (avance 2)	1.0
Lectura Lectura comprensiva de los temas de la unidad 8 en los siguientes recursos: Bibliografía básica [1] Unidades: Recursividad	2.0						
Aprendizaje Basado en Proyectos semana 12 Aprendizaje Basado en Proyectos Actividad: Modularizar el código en funciones; implementar procesos de comparación o análisis	1.0						
Estudio de caso práctico Implementación en lenguaje de alto nivel de problemática dada (avance 2)	1.0						
Horas del componente: Aprendizaje autónomo	4.0						

Semana 13

Resultados de aprendizaje de la asignatura	<ul style="list-style-type: none"> Analiza y explica el comportamiento de programas que involucren estructuras fundamentales de programación como variables, expresiones, asignaciones, archivos, estructuras de decisión y control, arreglos, funciones, paso de parámetros, y recursividad Construye, ejecuta y depura programas usando un IDE moderno Aplica estándares de documentación y estilo de programación que contribuyan a la legibilidad y mantenimiento de software
Contenidos a desarrollarse	<p>Unidad 7 / Unidad 8</p> <p>Unidad 7: Recursividad</p> <ul style="list-style-type: none"> Introducción y conceptos básicos de recursividad. Recursividad vs Iteración Tipos de recursividad Aplicación de la recursividad <p>Unidad 8: Librerías, API's de desarrollo.</p> <ul style="list-style-type: none"> Creación de paquetes y librerías en lenguajes de alto nivel. Consumo de API's de lenguajes de programación de alto nivel.
Actividades del componente: Aprendizaje en contacto con el docente	<ul style="list-style-type: none"> Clase magistral de contenidos <p>Exposición y explicación de contenidos</p> <ul style="list-style-type: none"> Resolución de inquietudes <p>Respuesta a preguntas por parte de los estudiantes, sobre las temáticas estudiadas.</p> <ul style="list-style-type: none"> Resolución de problemas <p>Planteamiento y guía en la solución de problemas</p> <ul style="list-style-type: none"> Tutorías <p>Tutorías</p> <ul style="list-style-type: none"> Evaluación del parcial del segundo bimestre <p>Resolución de las preguntas y problemas de la evaluación parcial del segundo bimestre por parte de los estudiantes.</p>
Horas del componente: Aprendizaje en contacto con el docente	3.00
Actividades del componente: Aprendizaje práctico - experimental	<ul style="list-style-type: none"> Taller 13 - Grupal (Laboratorio) <p>Construcción de programas usando paquetes y API's de desarrollo</p> <ul style="list-style-type: none"> Taller 12 Individual (Laboratorio) <p>Construcción de programas usando recursividad</p>
Horas del componente: Aprendizaje práctico	2.0

- experimental	
Actividades del componente: Aprendizaje autónomo	<div> Lectura Lectura comprensiva de los temas de la Unidad 9 en los siguientes recursos: <ol style="list-style-type: none"> 1. Bibliografía básica [1] Unidades: Manejo de excepciones: un análisis más detallado. 2. Bibliografía básica [2] Unidades: Excepciones: Lanzamiento, intercepción y tratamiento. </div> 2.0
	<div> Estudio de caso práctico Implementación en lenguaje de alto nivel de problemática dada (avance 3) </div> 2.0
Horas del componente: Aprendizaje autónomo	4.0

Semana 14

Resultados de aprendizaje de la asignatura	<ul style="list-style-type: none"> • Analiza y explica el comportamiento de programas que involucran estructuras fundamentales de programación como variables, expresiones, asignaciones, archivos, estructuras de decisión y control, arreglos, funciones, paso de parámetros, y recursividad • Identifica y previene errores de codificación comunes a nivel lógico y de sintaxis, a través del uso de excepciones • Aplica estrategias para la prueba y depuración de programas • Construye, ejecuta y depura programas usando un IDE moderno • Aplica estándares de documentación y estilo de programación que contribuyan a la legibilidad y mantenimiento de software
Contenidos a desarrollarse	Unidad 9: Programación a la defensiva. <ul style="list-style-type: none"> • Conceptos de Programación Defensiva • Manejo de Assertions • Introducción al manejo de excepciones • Estrategias para el Manejo de Excepciones • Diseño de excepciones propias • Ejemplificación de excepciones en lenguajes de alto nivel
Actividades del componente: Aprendizaje en contacto con el docente	<ul style="list-style-type: none"> • Clase magistral de contenidos Exposición y explicación de contenidos • Resolución de problemas Planteamiento y guía en la solución de problemas • Tutorías Tutorías

Horas del componente: Aprendizaje en contacto con el docente	3.0						
Actividades del componente: Aprendizaje práctico - experimental	<ul style="list-style-type: none"> Taller 14 Individual (Laboratorio) <p>Construcción de programas usando Excepciones</p>						
Horas del componente: Aprendizaje práctico - experimental	2.0						
Actividades del componente: Aprendizaje autónomo	<table border="1"> <tr> <td> Aprendizaje Basado en Proyectos semana 14 Aprendizaje Basado en Proyectos Actividad: Agregar mejoras opcionales: reportes, exportación de datos </td><td>1.0</td></tr> <tr> <td> Estudio de caso práctico Implementación en lenguaje de alto nivel de problemática dada (avance 4) </td><td>1.0</td></tr> <tr> <td> Lectura Lectura comprensiva de los temas de la Unidad 10 en los siguientes recursos: Bibliografía básica [1] Unidades: Archivos, flujos y serialización de objetos Bibliografía básica [17] Unidades: Registros y archivos </td><td>2.0</td></tr> </table>	Aprendizaje Basado en Proyectos semana 14 Aprendizaje Basado en Proyectos Actividad: Agregar mejoras opcionales: reportes, exportación de datos	1.0	Estudio de caso práctico Implementación en lenguaje de alto nivel de problemática dada (avance 4)	1.0	Lectura Lectura comprensiva de los temas de la Unidad 10 en los siguientes recursos: Bibliografía básica [1] Unidades: Archivos, flujos y serialización de objetos Bibliografía básica [17] Unidades: Registros y archivos	2.0
Aprendizaje Basado en Proyectos semana 14 Aprendizaje Basado en Proyectos Actividad: Agregar mejoras opcionales: reportes, exportación de datos	1.0						
Estudio de caso práctico Implementación en lenguaje de alto nivel de problemática dada (avance 4)	1.0						
Lectura Lectura comprensiva de los temas de la Unidad 10 en los siguientes recursos: Bibliografía básica [1] Unidades: Archivos, flujos y serialización de objetos Bibliografía básica [17] Unidades: Registros y archivos	2.0						
Horas del componente: Aprendizaje autónomo	4.0						

Semana 15

Resultados de aprendizaje de la asignatura	<ul style="list-style-type: none"> Analiza y explica el comportamiento de programas que involucran estructuras fundamentales de programación como variables, expresiones, asignaciones, archivos, estructuras de decisión y control, arreglos, funciones, paso de parámetros, y recursividad Identifica y previene errores de codificación comunes a nivel lógico y de sintaxis, a través del uso de excepciones Aplica estrategias para la prueba y depuración de programas Construye, ejecuta y depura programas usando un IDE moderno Aplica estándares de documentación y estilo de programación que contribuyan a la legibilidad y mantenimiento de software Extrae y manipula información de archivos de datos
---	--

Contenidos a desarrollarse	Unidad 10: Archivos <ul style="list-style-type: none"> • Escritura de archivos • Lectura de archivos 	
Actividades del componente: Aprendizaje en contacto con el docente	<ul style="list-style-type: none"> • Clase magistral de contenidos Exposición y explicación de contenidos • Resolución de problemas Planteamiento y guía en la solución de problemas • Tutorías Tutorías 	
Horas del componente: Aprendizaje en contacto con el docente	3.0	
Actividades del componente: Aprendizaje práctico - experimental	<ul style="list-style-type: none"> • Taller 15 Grupal (Laboratorio) Construcción de programas usando lectura y escritura de archivos 	
Horas del componente: Aprendizaje práctico - experimental	2.0	
Actividades del componente: Aprendizaje autónomo	Revisión de contenidos Revisión de contenidos de las unidades como preparación de la evaluación final del segundo bimestre.	3.0
	Estudio de caso práctico Implementación en lenguaje de alto nivel de problemática dada (avance 5)	1.0
Horas del componente: Aprendizaje autónomo	4.0	

Semana 16		
	<ul style="list-style-type: none"> • Analiza y explica el comportamiento de programas que involucren estructuras fundamentales de programación como variables, expresiones, asignaciones, archivos, estructuras de decisión y control, arreglos, funciones, paso de parámetros, y recursividad 	

Resultados de aprendizaje de la asignatura	<ul style="list-style-type: none">• Diseña, implementa, prueba, y depura un programa que usa estructuras fundamentales de programación• Identifica y previene errores de codificación comunes a nivel lógico y de sintaxis, a través del uso de excepciones• Aplica estrategias para la prueba y depuración de programas• Construye, ejecuta y depura programas usando un IDE moderno• Aplica estándares de documentación y estilo de programación que contribuyan a la legibilidad y mantenimiento de software• Extrae y manipula información de archivos de datos		
Contenidos a desarrollarse	Evaluación del segundo bimestre		
Actividades del componente: Aprendizaje en contacto con el docente	<ul style="list-style-type: none">• Evaluación del segundo bimestre <p>Explicación y resolución de inquietudes durante evaluación bimestral</p> <ul style="list-style-type: none">• Tutorías <p>Tutorías</p>		
Horas del componente: Aprendizaje en contacto con el docente	3.0		
Actividades del componente: Aprendizaje práctico - experimental	<ul style="list-style-type: none">• Resolución de ejercicios <p>Resolución y guía de ejercicios y preparación para evaluación de recuperación</p>		
Horas del componente: Aprendizaje práctico - experimental	2.0		
Actividades del componente: Aprendizaje autónomo	Preparación para la evaluación de recuperación		1.0
	Planteamiento de problemáticas que permitan la preparación para la evaluación de recuperación		
	Aprendizaje Basado en Proyectos semana 16		2.0
Aprendizaje Basado en Proyectos			
Actividad: Exposición final: aplicación completa más entrega de código final y documentación		1.0	
Estudio de caso práctico			
Implementación en lenguaje de alto nivel de problemática dada (presentación y defensa)			
Horas del componente:			

FECHAS IMPORTANTES Segundo Bimestre:

- Proyecto de investigación: Semana 10 - del 08/12/2025 al 14/12/2025
- Prácticas: Semana 12 - del 22/12/2025 al 04/01/2026
- Proyecto de investigación: Semana 12 - del 22/12/2025 al 04/01/2026
- Prueba parcial: Semana 13 - del 05/01/2026 al 11/01/2026
- Proyecto de investigación: Semana 14 - del 12/01/2026 al 18/01/2026
- Examen bimestral: Semana 16 - del 26/01/2026 al 31/01/2026
- Proyecto de investigación: Semana 16 - del 26/01/2026 al 31/01/2026
- Cuaderno de trabajo: Semana 16 - del 26/01/2026 al 31/01/2026

EVALUACIÓN DEL Segundo Bimestre

Componente	Porcentaje	Actividad	Instrumento de evaluación	Tipo de evaluación	Modalidad de evaluación	Ponderación
Aprendizaje en contacto con el docente (3.5 puntos)	35%	Prueba parcial	Cuestionario	Sumativa	H eteroevaluación	10.00 %
		Examen bimestral	E valuaciones	Sumativa	H eteroevaluación	25.00 %
Aprendizaje práctico - experimental (3.5 puntos)	35%	Taller grupal	Resolución de ejercicios	Sumativa	H eteroevaluación	15.00 %
		Prácticas		F ormativa		
		Taller individual	Resolución de ejercicios	Sumativa	H eteroevaluación	20.00 %
Aprendizaje autónomo (3.0 puntos)	30%	Proyecto de investigación	Resolución d e problemas	Sumativa	H eteroevaluación	20.00 %
		Cuaderno de trabajo	Resolución d e problemas	Sumativa	H eteroevaluación	10.00 %

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 70% de la nota (7 puntos). Esta calificación será sumada a lo acumulado por el estudiante en el componente de “Aprendizaje autónomo”.

F. DATOS BÁSICOS DEL DOCENTE:

Nombre del docente	ANGEL EDUARDO ENCALADA ENCALADA
Título de cuarto nivel:	Máster en Análisis y Visualización de Datos Masivos / Visual Analytics and Big Data
Departamento:	CIENCIAS DE LA COMPUTACIÓN Y ELECTRÓNICA
Correo electrónico:	aeencalada@utpl.edu.ec
Teléfono(Extensión):	073701444 - Ext: 2532
Currículo profesional resumido:	Ingeniero en Informática por la Universidad Técnica Particular de Loja (UTPL, Ecuador) y Máster en Análisis y Visualización de Datos Masivos / Visual Analytics and Big Data por la Universidad Internacional de La Rioja (España). Actualmente es miembro del Departamento de Ciencias de la Computación y Electrónica de la UTPL, donde se desempeña como docente en las carreras de Computación y Tecnologías de la Información. Su actividad docente abarca asignaturas relacionadas con programación y lógica computacional, diseño y administración

de bases de datos, inteligencia de negocios y análisis de datos. Cuenta con experiencia en administración de bases de datos, analítica y visualización de información, análisis y diseño de sistemas de información, así como en gestión de infraestructura tecnológica.

Nombre del docente	PEDRO DANIEL IRENE ROBALINO
Título de cuarto nivel:	Master Universitario en Ingeniería de Software para la Web
Departamento:	CIENCIAS DE LA COMPUTACIÓN Y ELECTRÓNICA
Correo electrónico:	pdirene@utpl.edu.ec
Teléfono(Extensión):	073701444 - Ext: 2527
Currículo resumido: profesional	Ingeniero en Sistemas Informáticos y Computación por la Universidad Técnica Particular de Loja - Ecuador. Maestría en Ingeniería de Software para la Web por la Universidad de Alcalá de Henares - España. Docente Universitario en la UTPL en asignaturas de Programación, Arquitectura de Aplicaciones, Cálculo, Matemática por más de 10 años. Labora en el Departamento de Ciencias de la Computación y Telecomunicaciones. Miembro del Equipo de calidad de la carrera.

Nombre del docente	RENE ROLANDO ELIZALDE SOLANO
Título de cuarto nivel:	Máster Universitario en Ingeniería del Software para la Web - Universidad de Alcalá.
Departamento:	CIENCIAS DE LA COMPUTACIÓN Y ELECTRÓNICA
Correo electrónico:	rrelizalde@utpl.edu.ec
Teléfono(Extensión):	073701444 - Ext: 2533
Currículo resumido: profesional	Máster Universitario en Ingeniería del Software para la Web - Universidad de Alcalá. Ingeniero en Sistemas - Universidad Nacional de Loja. Docente Investigador del Departamento de Ciencias de la Computación y Electrónica. 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. Docente invitado a la maestría de Ingeniería del Software de la Universidad Nacional de Loja.

G. RECURSOS A UTILIZAR EN EL DESARROLLO DE LA ASIGNATURA

BIBLIOGRAFÍA BÁSICA:

Nombre del texto	CÓMO PROGRAMAR EN JAVA,	
Citado según normas APA vigente	Como programar en Java / Paul J. Deitel y Hervey Deitel. (2016). Pearson. https://visorweb.utpl.edu.ec/library/publication/como-programar-en-java-1617638467	
Importancia para el estudio de la asignatura	<p>Se ha seleccionado este texto por las ventajas pedagógicas y técnicas que brinda al estudiante; se tratan temas relacionados con el lenguaje de alto nivel Java desde conceptos, estructuras, sintaxis básicas, hasta el uso de tecnologías avanzadas. De forma tal que el recurso servirá al estudiante en el transcurso de la carrera. Adicionalmente en el texto se puede encontrar una gran cantidad de ejercicios los cuales son explicados a detalle por parte de los autores.</p>	
Disponible en biblioteca	Física	Virtual
	x	x

BIBLIOGRAFÍA COMPLEMENTARIA:

Nombre del texto	[4] Fernández, Arturo (2013). Python 3 al descubierto. Segunda Edición. Rc Libros. https://visorweb.utpl.edu.ec/library/publication/python-3-al-descubierto-1604103079
La obra ofrece una visión general de los conceptos del lenguaje de programación Python. Indica a los lectores los pasos iniciales para la instalación y puesta en marcha de ejercicios de programación haciendo uso de las características y ventajas del lenguaje de programación Python.	
Nombre del texto	[17] López Román, Leobardo. (2013) Metodología de la programación orientada a objetos. México. https://visorweb.utpl.edu.ec/library/publication/metodologia-de-la-programacion-orientada-a-objetos-1603157739
El texto de Metodología de la programación orientada a objetos permite estudiar de forma teórica y práctica los conceptos de la programación. Las temáticas que se abordan son: fundamentos básicos de la programación, la secuenciación, la selección, ciclos repetitivos y métodos.	
Nombre del texto	[2] Blasco, F. (2019). Programación orientada a objetos en Java. Ediciones de la U. https://elibro.net/es/lc/bibliotecaupl/titulos/127125
En la presente obra se focaliza en exponer los recursos teóricos y prácticos necesarios para aprender sobre los fundamentos de programación. Se lo realiza a través del lenguaje de alto nivel Java. Para la asignatura de Introducción a la Programación, se	

solicita revisar los capítulos: Tipos de Datos, Operadores, Expresiones; Estructuras de Control; Excepciones: Lanzamientos, Intercepción y Tratamiento.

Nombre del texto	[3] Cadenhead, Roger (2014). PROGRAMACIÓN JAVA 8. Anaya
La obra cuenta con un sencillo enfoque paso a paso para desarrolladores de todos los niveles que requieran aprender a programar en lenguaje de programación Java. Se explica los ejemplos a través de la herramienta de desarrollo que brinda el IDE de programación Netbeans.	

Nombre del texto	[6] Thomas, Erl (2013). Cloud computing: concepts, technology & architecture. New Jersey. Prentice Hall
El libro permite estudiar las bases teóricas y prácticas para el uso de Cloud Computing en el planteamiento de soluciones informáticas.	

REA	Enlace
10. Pro Git	https://git-scm.com/book/es/v2
11. Java Apuntes Básicos	http://j4loxa.com/courses/java101/Java_Apuntes_Basicos.epub
12. Introducción a Java	https://campusvirtual.ull.es/ocw/course/view.php?id=139
13. Python Documentación	https://docs.python.org/3/
14. Java Documentación	https://www.oracle.com/technetwork/es/java/javase/documentation/index.html
15. Docker documentación	https://www.docker.com/
16. Git Documentación	https://git-scm.com/
7. Introduction to Computer Science and Programming.	https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-00sc-introduction-to-computer-science-and-programming-spring-2011/
8. Introduction to Computer Science and Programming in Python.	https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-0001-introduction-to-computer-science-and-programming-in-python-fall-2016/
9. Del cloud computing al big data.	https://campusvirtual.ull.es/ocw/pluginfile.php/9374/mod_resource/content/4/Del_Cloud_Computing.al_Big_Data_JordiTorres.ES_.pdf

H. ELABORACIÓN Y APROBACIÓN

	Nombre	Firma
<i>Elaborado por:</i>	RENE ROLANDO ELIZALDE SOLANO	
<i>Fecha de elaboración:</i>	02-10-2025	
<i>Revisado por:</i>	IRMA ELIZABETH CADME SAMANIEGO	
<i>Aprobado por:</i>	FERNANDA MARICELA SOTO GUERRERO	
	Fecha de aprobación:	03-10-2025