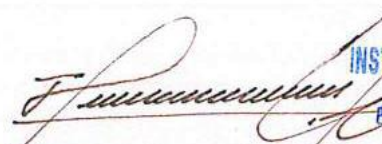





Programa de Estudios de la Unidad de Aprendizaje: PROGRAMACIÓN BÁSICA																										
Clave: 3FP-FM977				Créditos: 3.37				Programa Académico: TÉCNICO EN PROGRAMACIÓN																		
								Nivel:				1°	2°	3°	4°	5°	6°									
Ramas de Conocimiento										Unidades Académicas donde se Imparte:																
Ingeniería y Ciencias Físico Matemáticas	X	Ciencias Sociales Administrativas		Ciencias Médico Biológicas		TODAS LAS U.A.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	CET1
Área de Formación Curricular										Tiempos Asignados:																
Institucional		Científica, Humanística y Tecnológica Básica		Profesional	X	Global: 54 Hrs/18 semanas/Semestre		Aula: 1 Hrs/Semana		Total: 18 Hrs/Semestre		Taller: 0 Hrs/Semana		Total: 0 Hrs/Semestre		Laboratorio: 2 Hrs/Semana		Total: 36 Hrs/Semestre		Otros ambientes de aprendizaje: 0 Hrs/Semana		Total: 0 Hrs/Semestre				
Tipo de Espacio																										
Aula	X	Taller		Laboratorio	X																					
Modalidad																										
Escolarizada	X	No Escolarizada		Mixta																						
Vigencia a Partir:				AGOSTO 2023																						
Proceso de Diseño y Autorización:					Día	Mes	Año	Por Unidad de Aprendizaje:		X	Por Área:		Por Módulo:		Firma y Sello de Autorización:											
Elaborado por:	REP. ACAD. NMS	Fecha de Elaboración:	04	02	2022									  <p><b>M. EN E.N.A MARÍA ISABEL ROJAS RUÍZ</b> Directora de Educación Media Superior</p>												
Revisado por:	DEMS	Fecha de Revisión:	10	01	2023																					
Aprobado por:	CTCE-NMS	Fecha de Aprobación:	12	01	2023																					
Autorizado por:	CPA-CGC	Fecha de Autorización:	16	02	2023																					



## Programa Académico: Técnico en Programación

## Unidad de Aprendizaje: Programación Básica

### FUNDAMENTACIÓN

Enseguida se presentan los argumentos que exponen las bases epistémicas y didácticas, así como la relevancia del Programa de Estudios de la unidad de aprendizaje **Programación Básica** para la formación del estudiante.

La unidad de aprendizaje **Programación Básica** pertenece al área de formación profesional del Bachillerato Tecnológico Bivalente del Nivel Medio Superior del Instituto Politécnico Nacional, se ubica en el tercer nivel del Plan de Estudios del Programa Académico Técnico en Programación y se imparte en la modalidad escolarizada, de manera obligatoria en la rama del conocimiento de Ingeniería y Ciencias Físico-Matemáticas.

Esta unidad de aprendizaje coadyuva a comprender la **Programación básica** como una dimensión científica, técnica, tecnológica, social, responsable, metodológica y sustentable, que incentiva la adquisición, desarrollo y aplicación del pensamiento lógico, analítico, el razonamiento abstracto, la creatividad, la iniciativa, la resolución a problemas y diversas habilidades cognitivas. Introduce al estudiante al campo conceptual, procedimental y actitudinal para escribir programas en un lenguaje de programación, resolver problemas de programación básica, diseñar, desarrollar y comprender los complejos problemas computacionales básicos para solucionarlos de una manera efectiva, considerando los principios y dimensiones del desarrollo humano sustentable como la perspectiva de género. La adquisición de estas destrezas y habilidades relacionadas con el pensamiento analítico favorecerán en el estudiante el desarrollo de una visión crítica y holística, cuya puesta en práctica, en forma autónoma, en el futuro le coadyuvará a responder en forma eficiente y eficaz a los retos que se le presenten al incorporarse a estudios superiores o de igual forma al campo laboral.

La unidad de aprendizaje **Programación Básica** contribuye al desarrollo del talento requerido por la Industria 4.0 y para la transformación del país, orientada al logro del desarrollo humano sustentable de México. Esto debido a que adquirirá, desarrollará y aplicará conceptos, técnicas y métodos que favorecen el análisis, desarrollo y resolución de problemas básicos computacionales, mediante la escritura de programas en un lenguaje de programación, como instrumento estructural para resolver, organizar y analizar datos en forma lógica, representarlos mediante abstracciones, desde los diagramas de flujo, diseñar y escribir programas para dar solución a problemas cotidianos mediante el pensamiento analítico, crítico y algorítmico, optimizar e innovar soluciones extrapolando en una solución informática en un lenguaje de programación.

**Programación Básica** es una unidad de aprendizaje enfocada al desarrollo de habilidades técnicas, cognitivas y socioemocionales inherentes al estudio, análisis, aplicación y resolución de problemas, mediante la escritura de programas en un lenguaje de programación, con la finalidad de resolver problemas computacionales básicos.

La unidad de aprendizaje **Programación Básica** estará fundamentada en el Modelo Educativo Institucional vigente y en la Educación para la industria 4.0, por esto, se empleará la metodologías didáctica activas como el Aprendizaje Basado en Problemas, o el Aprendizaje Basado en Retos, entre otras; esto con el propósito de que el estudiante desarrolle competencias de un entorno 4.0, como el trabajo colaborativo, trabajo en equipo, reto al cambio, autodirección, resolución de problemas cercanos a la realidad, autogestión del aprendizaje y resiliencia. Además, se emplearán herramientas tecnológicas que fomentarán la colaboración e interacción presenciales y virtuales, en forma síncrona o asíncrona, que corresponden a la Educación 4.0. También se emplearán aplicaciones informáticas para el análisis, diseño, codificación y pruebas de problemas computacionales comunes, entre otros.

El rol del profesor será de mediador entre el estudiante y los contenidos didácticos a abordar, puesto que se centrará en la creación, organización, supervisión y mediación de los espacios de trabajo, incluidas las aulas virtuales, el aula invertida, los ciberespacios, sin dejar de atender las necesidades técnicas, de conocimientos, apoyo logístico y metodológico en los procesos de aprendizaje individual y grupal, con el objetivo de generar ambientes que favorezcan la educación técnica, inclusiva, flexible, sustentable y con perspectiva de género.







## Programa Académico: Técnico en Programación

## Unidad de Aprendizaje: Programación Básica

El estudiante desarrollará un trabajo autónomo en diferentes ambientes de aprendizaje, organizará el trabajo de manera independiente y articulará saberes de diversos campos del conocimiento, que le posibilitarán construir y expresar su propio conocimiento en beneficio de la sociedad; también adquirirá habilidades tanto tecnológicas como personales que promoverán la comunicación asertiva, la creatividad, la negociación, la gestión del tiempo, la motivación, el liderazgo y la responsabilidad social vinculada a la protección del medio ambiente, la erradicación de toda manifestación de violencia de género, la inclusión y la accesibilidad.

La evaluación se efectuará en el marco de la evaluación auténtica, por esto, comprenderá tres momentos: diagnóstica, formativa y sumativa. La evaluación diagnóstica se llevará a cabo mediante una charla informal sobre conocimientos previos con evaluación y retroalimentación durante el mismo momento, con la finalidad de que el docente efectúe los ajustes didácticos pertinentes y de ser necesario, nivele y ajuste los conocimientos previos adquiridos en otras unidades de aprendizaje para que establezca conexiones significativas con las unidades didácticas de la unidad de aprendizaje. Un segundo momento de la evaluación hace referencia a la evaluación formativa, que se desarrollará a lo largo del proceso de enseñanza-aprendizaje mediante las secuencias didácticas y actividades de aprendizaje formativas que estimulen el aprendizaje activo y significativo del estudiante. Este momento se enriquecerá con diversos tipos de evaluación, como la autoevaluación, la coevaluación y heteroevaluación, puesto que coadyuvarán a dar seguimiento al desarrollo de los saberes y habilidades en contexto. Cabe señalar que estas clases de evaluación serán reforzadas a través de la retroalimentación efectiva y constante.

En el tercer momento de la evaluación, con fines de acreditación, se diseñarán escenarios con problemas comunes que permitan recuperar el nivel de logro y conducir al estudiante a la meta cognición en la unidad de aprendizaje **Programación Básica**, mediante evidencias de conocimiento, comprensión y diseño de algoritmos, escritura de programas en algún lenguaje de programación, compilación de los programas escritos, entre otras evidencias de aprendizaje, cuyos criterios, aspectos e indicadores serán conocidos por los estudiantes en forma previa. Las evidencias de evaluación formativa e integradora mostrarán el saber hacer de manera reflexiva de los estudiantes, utilizando el conocimiento que van adquiriendo durante el proceso didáctico para luego transferir y aplicar este aprendizaje en contextos escolares, sociales y laborales.

En base a la flexibilidad curricular y en el reconocimiento de aprendizajes múltiples, también podrá aplicarse una evaluación general para verificar que los conocimientos adquiridos por el estudiante demuestren que domina los saberes, objetivos y alcances de la **Programación Básica**, al finalizar el periodo ordinario. De esa forma, el programa de estudios de esta unidad de aprendizaje establece estándares para el desarrollo de conocimientos, habilidades socioemocionales, actitudes y valores.





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

**DESCRIPCIÓN DE LA UNIDAD DE APRENDIZAJE**

Unidad de Aprendizaje: Programación Básica		
Propósito de la Unidad de Aprendizaje		
<p>Construye programas informáticos en un lenguaje de programación, para resolver problemas de informática, con base en los algoritmos, en las estructuras de selección, repetición, en los arreglos unidimensionales, multidimensionales, en la ordenación y búsqueda en listas de forma analítica, creativa e innovadora, considerando un enfoque de desarrollo practico y sustentable.</p>		
Unidad 1: Introducción a la programación		
Unidad de competencia	Aprendizajes esperados	Contenidos de aprendizaje
<p>1.Relaciona los conceptos y paradigmas de programación, utilizando herramientas para el aprendizaje de los procesos de compilación y ejecución de los programas con base en un entorno de desarrollo integrado (IDE) de forma innovadora, eficiente y creativa.</p>	<p>1. Domina el concepto de la programación, mediante la utilización de sus características y elementos principales, con base al pensamiento lógico y a la solución a diversos problemas y paradigmas de la programación de una forma individual y creativa.</p>	<p><b>Conceptual:</b> Conceptos básicos de programación Datos, tipos de datos y operaciones primitivas Paradigmas de programación <b>Procedimental:</b> Utiliza las herramientas de programación básicas incluidas en el sistema operativo para la ejecución de programas. <b>Actitudinal:</b> Atiende una actitud respetuosa hacia sus compañeros y docentes, el desarrollo de sus actividades las realiza de una forma analítica, reflexiva, con una comunicación efectiva y el trabajo se cumple de forma colaborativa.</p>
	<p>2.Ejecuta programas mediante un editor y compilador independientes, en un entorno de desarrollo integrado (IDE) para la resolución de problemas diversos con un enfoque de innovación, eficiente y creativo.</p>	<p><b>Conceptual:</b> Entorno de desarrollo IDE Operación de asignación Estructura de programas <b>Procedimental:</b> Utiliza de manera eficiente el entorno de desarrollo integrado (IDE) para la elaboración de los programas informáticos. <b>Actitudinal:</b> Atiende una actitud respetuosa hacia sus compañeros y docentes, el desarrollo de sus actividades las realiza de una forma analítica, reflexiva, con una comunicación efectiva y el trabajo se cumple de forma colaborativa.</p>
Unidad 2: Elementos básicos de la programación		
Unidad de competencia	Aprendizajes esperados	Contenidos de aprendizaje
<p>2.Desarrolla programas considerando los elementos básicos que lo componen,</p>	<p>1. Codifica programas utilizando estructuras de control condicionales y selectivas para aumentar su funcionalidad, mediante los tipos primitivos de</p>	<p><b>Conceptual:</b> Expresiones Reglas de prioridad Funciones internas</p>





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

<p>mediante operaciones de entrada - salida para resolver problemas comunes de informática con un enfoque creativo, de innovación y de trabajo en equipo.</p>	<p>datos, los operadores aritméticos y la conversión de tipos de datos, construyendo funciones de manera creativa, asertiva y con innovación</p>	<p>Conversión de tipo de datos El flujo de control de un programa Estructura secuencial Estructuras selectivas Alternativas Estructuras de decisión anidadas (en escalera) La sentencia ir-a (goto) Estructuras Repetitivas Estructuras Repetitivas anidadas <b>Procedimental:</b> Construye programas utilizando operadores, variables, constantes, tipos primitivos de datos, operadores aritméticos y conversión de tipos de datos, para resolver problemas comunes de informática. Ejecuta instrucciones que incluyen estructuras de condición, de selección, de repetición para resolver problemas comunes de informática. <b>Actitudinal:</b> Se interesa en la resolución de problemas comunes de informática de una manera creativa, asertiva e innovadora. Respeta el trabajo de sus compañeros y docentes permitiendo el desarrollo de las actividades de una manera colaborativa.</p>
	<p>2. Construye programas que implementen funciones para resolver problemas comunes de informática con un enfoque creativo, de innovación y de trabajo en equipo.</p>	<p><b>Conceptual:</b> Funciones Procedimientos (subrutinas) Ámbito: variables locales y globales Comunicación con subprogramas Funciones y procedimientos como parámetros Recursión (recursividad) Sobrecarga de funciones <b>Procedimental:</b> Crea funciones para resolver problemas comunes de informática Utiliza procedimientos y subrutinas para la comunicación entre programas y resolver problemas comunes de informática. Realiza programas que utilizan recursividad para resolver problemas comunes de informática. <b>Actitudinal:</b> Se interesa en la resolución de problemas comunes de informática de una manera creativa, asertiva e innovadora. Respeta el trabajo de sus compañeros y docentes permitiendo el desarrollo de las actividades de una manera colaborativa.</p>

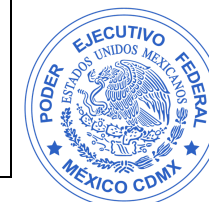




**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

Unidad 3: Arreglos y estructuras de datos		
Unidad de competencia	Aprendizajes esperados	Contenidos de aprendizaje
	1. Construye programas que utilizan arreglos unidimensionales y multidimensionales mediante las estructuras de datos, operación con vectores para solucionar y resolver problemas reales y cotidianos con innovación, adaptabilidad, empatía y proactividad.	<p><b>Conceptual:</b> Estructuras de datos Arrays (arreglos) unidimensionales: los vectores Operaciones con vectores Arrays de varias dimensiones Arrays multidimensionales Almacenamiento de arrays en memoria</p> <p><b>Procedimental:</b> Utiliza las estructuras de datos, operación con vectores para realizar ordenación y búsqueda de elementos. Aplica arrays unidimensionales y multidimensionales para resolver problemas reales y cotidianos.</p> <p><b>Actitudinal:</b> Respeta las ideas y opiniones de sus compañeros y docentes. Se interesa por la resolución de los problemas encontrados en su entorno con innovación, adaptabilidad, empatía y proactividad. Atiende las problemáticas de su entorno, trabajando con una comunicación asertiva y de forma empática.</p>
	2. Construye programas que utilizan uniones, enumeraciones, ordenación de listas y búsquedas secuenciales para solucionar y resolver problemas reales y cotidianos, con innovación, adaptabilidad, empatía y proactividad.	<p><b>Conceptual:</b> Estructuras versus registros Arrays de estructuras Uniones Unión versus estructura Enumeraciones Ordenación Búsqueda Intercalación</p> <p><b>Procedimental:</b> Utiliza uniones versus estructuras para la resolución de problemas informáticos comunes. Realiza programas que utilizan la ordenación y las búsquedas para resolver problemas reales y cotidianos de su entorno.</p> <p><b>Actitudinal:</b> Respeta las ideas y opiniones de sus compañeros y docentes. Se interesa por la resolución de los problemas encontrados en su entorno con innovación, adaptabilidad, empatía y proactividad. Atiende las problemáticas de su entorno, trabajando con una comunicación asertiva y de forma empática.</p>
3. Desarrolla programas utilizando arreglos o arrays mediante la manipulación de estructuras básicas de datos para ordenación y búsqueda, de manera analítica, eficiente y creativa.		





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

➔ **MATRIZ DE VINCULACIÓN** ⬅

	Unidad de Competencia 1			Unidad de Competencia 2			Unidad de Competencia 3		
	AE 1	AE 2	AE 3	AE 1	AE 2	AE 3	AE 1	AE 2	AE 3
<b>COMPETENCIAS PARA EL SIGLO XXI HABILIDADES BLANDAS Y SOCIOEMOCIONALES</b>									
Trabajo en equipo							X	X	X
Manejo de la frustración		X	X	X	X	X	X	X	X
Creatividad				X	X	X	X	X	X
Pensamiento crítico	X	X	X	X	X	X	X	X	X
Inteligencia emocional			X	X	X	X	X	X	X
Innovación			X	X	X	X	X	X	X
Liderazgo							X	X	X
Adaptabilidad			X	X	X	X	X	X	X
Proactividad		X	X	X	X	X	X	X	X
Empatía		X	X	X	X	X	X	X	X
Comunicación asertiva			X	X	X	X	X	X	X





## Programa Académico: Técnico en Programación

## Unidad de Aprendizaje: Programación Básica

### PERFIL DOCENTE

El profesor que imparta la Unidad de Aprendizaje **Programación Básica** contará con las habilidades en el manejo de los saberes disciplinares y/o profesionales, así como su disposición, autoridad y tolerancia en el manejo de grupos de aprendizaje. Por lo tanto, debe poseer las habilidades que favorezcan el desarrollo del talento 4.0.

#### Habilidades docentes en el desarrollo del Talento

En el campo de su especialización:

- Habilidades y conocimientos profesionales que se requiere para la impartición de la Unidad de Aprendizaje.
- Emplea habilidades digitales, desarrollarlas y actualizadas para el desarrollo de la Unidad de Aprendizaje.

En el campo pedagógico:

- Fomentar procesos de enseñanza que le permitan interpretar y resolver las necesidades de aprendizaje de los alumnos, tomando en cuenta sus capacidades, habilidades, vocación e intereses.
- Desarrollar procesos de enseñanza aprendizaje, utilizando métodos basados en administración de proyectos reales, aprovechando espacios educativos distintos a las aulas, para mejorar la calidad y pertinencia de la enseñanza.

En el campo de la investigación:

- Fortalecer el trabajo académico a partir del aprovechamiento de los resultados y productos de los proyectos de investigación

#### Perfil Profesional

- Licenciado o Ingeniero Titulado en Comunicaciones y Electrónica, en Computación o Informática o Maestría en ciencias computacionales o sistemas informáticos con experiencia de dos años en el área docente
- Experiencia comprobable de dos años en la iniciativa pública o privada aplicando los conocimientos de la unidad de aprendizaje

Se requiere de **tres auxiliares** para poder supervisar los ejercicios, programas, aplicaciones de software, revisando que se encuentren los elementos solicitados como conceptos básicos de programación, estructuras de control, de datos, de cada trabajo de cada alumno, y lleguen al aprendizaje significativo. Con cada ejercicio, práctica que realicen en laboratorio de esta unidad de aprendizaje irán adquiriendo las competencias con el apoyo de los **tres auxiliares** para solucionar dudas, guiar al estudiante apoyando **al titular**.





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

**ESTRUCTURA DIDÁCTICA**

<b>Unidad Didáctica 1:</b>	Introducción a la programación	<b>Nivel:</b>	3er Nivel
<b>Propósito General:</b>	Construye programas informáticos en un lenguaje de programación, para resolver problemas de informática, con base en los algoritmos, en las estructuras de selección, repetición, en los arreglos unidimensionales, multidimensionales, en la ordenación y búsqueda en listas de forma analítica, creativa e innovadora, considerando un enfoque de desarrollo práctico y sustentable.		
<b>Unidad de Competencia No 1:</b>	Relaciona los conceptos y paradigmas de programación, utilizando herramientas para el aprendizaje de los procesos de compilación y ejecución de los programas con base en un entorno de desarrollo integrado (IDE) de forma innovadora, eficiente y creativa.		
<b>Aprendizaje Esperado No 1:</b>	Domina el concepto de la programación, mediante la utilización de sus características y elementos principales, con base al pensamiento lógico y a la solución a diversos problemas y paradigmas de la programación de una forma individual y creativa.	<b>Tiempo estimado para obtener el Aprendizaje Esperado:</b>	9 horas

**Contenidos de Aprendizaje**

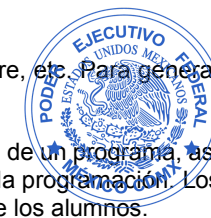
Conceptuales	Procedimentales	Actitudinales
<ol style="list-style-type: none"> <li>1. Conceptos básicos de programación               <ol style="list-style-type: none"> <li>1.1. Partes constitutivas de un programa</li> <li>1.2. Instrucciones y tipos de instrucciones</li> <li>1.3. Elementos básicos de un programa</li> </ol> </li> <li>2. Datos, tipos de datos y operaciones primitivas               <ol style="list-style-type: none"> <li>2.1. Constantes y variables</li> <li>2.2. Declaración de constantes y variables</li> </ol> </li> <li>3. Paradigmas de la programación               <ol style="list-style-type: none"> <li>3.1. Programación modular</li> <li>3.2. Programación estructurada</li> <li>3.3. Programación Orientada a objetos</li> </ol> </li> </ol>	<p>Utiliza las herramientas de programación básicas incluidas en el sistema operativo para la ejecución de programas.</p> <p>Organizador gráfico de los conceptos básicos y paradigmas de la programación.</p>	<p>Atiende una actitud respetuosa hacia sus compañeros y docentes, el desarrollo de sus actividades las realiza de una forma analítica, reflexiva, con una comunicación efectiva y el trabajo se cumple de forma colaborativa.</p>

**Estrategia Didáctica y Ambiente de Aprendizaje**

Se propone una metodología **basada en problemas**, para acompañar el proceso de enseñanza-aprendizaje dentro del aula.

**Apertura:** El docente, hace una exploración de conocimientos general, con una lluvia de ideas y conceptos sobre la programación, los sistemas, las tecnologías, el software, etc. Para generar con ello a partir de las respuestas y conocimientos previos del alumno sobre estos temas.

**Desarrollo:** El docente hace una presentación electrónica sobre los fundamentos y conceptos básicos de la programación, explica sus características las partes constitutivas de un programa, así como las instrucciones y los tipos de instrucciones. El docente explica los elementos básicos de programación y ejemplifica diversas problemáticas a ser resueltas mediante la programación. Los estudiantes buscan, discriminan y sintetizan información para construir su conceptualización de los elementos de programación, el docente retroalimenta las aportaciones de los alumnos.





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

**Cierre:** Los estudiantes responden un cuestionario sobre los conceptos básicos, las características y los tipos de instrucciones de un programa, así como anexan una breve reflexión sobre los elementos de programación, las partes de un programa y las instrucciones dentro del mismo, el docente está abierto a las preguntas o dudas que el alumno llegará a tener para facilitar su aprendizaje.

**Ambiente de Aprendizaje:**

Es un ambiente de aprendizaje presencial pero además el docente creará un aula virtual en alguna plataforma tecnológica como Teams, Classroom, etc. Para poder tener comunicación con los alumnos de manera asíncrona y con ello complementar la comunicación, resolver dudas y crear un ambiente híbrido (presencial y virtual).

Herramientas Tecnológicas y Recursos Didácticos	Evidencia de Aprendizaje para la Evaluación Formativa	Instrumento y Criterios de Evaluación
<p>Herramientas Tecnológicas:</p> <ul style="list-style-type: none"> <li>• Equipo de Cómputo (PC o Lap Top)</li> <li>• Conexión a internet</li> <li>• Video proyector</li> <li>• Software de Desarrollo de aplicaciones o IDE</li> <li>• Aulas Virtuales</li> </ul> <p>Recursos Didácticos:</p> <ul style="list-style-type: none"> <li>• Libros digitales o impresos relacionados al tema</li> <li>• Infografías y Vídeos relacionados al tema</li> <li>• Presentaciones electrónicas relacionadas al tema</li> <li>• Formularios interactivos relacionados al tema</li> <li>• Páginas, sitios web y aplicaciones móviles</li> </ul>	<p>Elaboración de organizador grafico de los conceptos básicos y paradigmas de la programación.</p>	<p>Instrumento de Evaluación:</p> <ul style="list-style-type: none"> <li>• Lista de cotejo</li> </ul> <p>Criterios de Forma (estilo):</p> <ul style="list-style-type: none"> <li>• Entrega de trabajo de investigación sin faltas de ortografía</li> <li>• Entrega de documento pdf en el aula y/o plataforma virtual</li> <li>• Entrega de actividad en tiempo y forma solicitados</li> </ul> <p>Criterios de Fondo (parte técnica):</p> <ul style="list-style-type: none"> <li>• Incluye los conceptos básicos de la programación.</li> <li>• Incluye los tipos de datos y las operaciones primitivas.</li> <li>• Incluye las características más importantes de los paradigmas de la programación.</li> </ul>





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

<b>Unidad Didáctica 1:</b>	Introducción a la programación	<b>Nivel:</b>	3er Nivel
<b>Propósito General:</b>	Construye programas informáticos en un lenguaje de programación, para resolver problemas de informática, con base en los algoritmos, en las estructuras de selección, repetición, en los arreglos unidimensionales, multidimensionales, en la ordenación y búsqueda en listas de forma analítica, creativa e innovadora, considerando un enfoque de desarrollo practico y sustentable.		
<b>Unidad de Competencia No 1:</b>	Relaciona los conceptos y paradigmas de programación, utilizando herramientas para el aprendizaje de los procesos de compilación y ejecución de los programas con base en un entorno de desarrollo integrado (IDE) de forma innovadora, eficiente y creativa.		
<b>Aprendizaje Esperado No 2:</b>	Ejecuta programas mediante un editor y compilador independientes, en un entorno de desarrollo integrado (IDE) para la resolución de problemas diversos con un enfoque de innovación, eficiente y creativo.	<b>Tiempo estimado para obtener el Aprendizaje Esperado:</b>	9 horas

**Contenidos de Aprendizaje**

Conceptuales	Procedimentales	Actitudinales
<ol style="list-style-type: none"> <li>1. El entorno de desarrollo integrado               <ol style="list-style-type: none"> <li>1.1. IDE Netbeans o similar</li> </ol> </li> <li>2. La operación de asignación               <ol style="list-style-type: none"> <li>2.1. Aritmética</li> <li>2.2. Lógica</li> <li>2.3. De cadenas de caracteres</li> <li>2.4. Múltiple</li> </ol> </li> <li>3. Estructura de programas               <ol style="list-style-type: none"> <li>3.1. Cabecera del programa</li> <li>3.2. Declaración de variables</li> </ol> </li> </ol>	<p>Utiliza de manera eficiente el entorno de desarrollo integrado (IDE) para la elaboración de los programas informáticos.</p> <p>Práctica 1 Programa “Hola Mundo” Práctica 2 Programa “Cálculo de área y perímetro de figura geométrica”</p>	<p>Atiende una actitud respetuosa hacia sus compañeros y docentes, el desarrollo de sus actividades las realiza de una forma analítica, reflexiva, con una comunicación efectiva y el trabajo se cumple de forma colaborativa.</p>

**Estrategia Didáctica y Ambiente de Aprendizaje**

Se propone una metodología **basada en problemas**, para acompañar el proceso de enseñanza-aprendizaje dentro del aula.

**Apertura:** El docente realiza un repaso de los elementos básicos de programación y ejemplifica diferentes problemáticas a ser resueltas mediante la programación.

**Desarrollo:** El docente explica y ejemplifica la creación, edición, compilación y ejecución de un programa que resuelva alguna problemática propuesta, en el editor incluido en el sistema operativo, posteriormente plantea problemas detonantes que den lugar a la aplicación de elementos de programación y orienta en la solución de problemas propuestos.

Los estudiantes resolverán en equipo dichos problemas mediante la creación, edición, compilación y ejecución de programas de software en el mismo editor incluido en el sistema operativo.

El docente explica conceptos fundamentales, elementos y características del entorno de desarrollo integrado, propone ejercicios donde los estudiantes instalen, configuren y utilicen el entorno de desarrollo integrado para que el estudiante siga los procedimientos de Instalación y configuración del mismo IDE.





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

**Cierre:** El docente resume los aprendizajes de los estudiantes fomentando a la corrección de errores más comunes en la codificación de programas e instalación y configuración del entorno de desarrollo IDE y les solicita que realicen primero de manera individual y después de forma colaborativa, la evidencia de aprendizaje formativa que son las practicas 1 y 2.

**Ambiente de Aprendizaje:**

Es un ambiente de aprendizaje presencial pero además el docente creará un aula virtual en alguna plataforma tecnológica como Teams, Classroom, etc. Para poder tener comunicación con los alumnos de manera asincrónica y con ello complementar la comunicación, resolver dudas y crear un ambiente híbrido (presencial y virtual).

Elaboración de Práctica 1 Programa “Hola Mundo”

Elaboración de Práctica 2 Programa “Cálculo de área y perímetro de figura geométrica”

Herramientas Tecnológicas y Recursos Didácticos	Evidencia de Aprendizaje para la Evaluación Formativa	Instrumento y Criterios de Evaluación
<p>Herramientas Tecnológicas:</p> <ul style="list-style-type: none"> <li>Equipo de Cómputo (PC o Lap Top)</li> <li>Conexión a internet</li> <li>Video proyector</li> <li>Software de Desarrollo de aplicaciones o IDE</li> <li>Aulas Virtuales</li> </ul> <p>Recursos Didácticos:</p> <ul style="list-style-type: none"> <li>Libros digitales o impresos relacionados al tema</li> <li>Infografías y Videos relacionados al tema</li> <li>Presentaciones electrónicas relacionados al tema</li> <li>Formularios interactivos relacionados al tema</li> <li>Páginas o sitios web y aplicaciones móviles</li> </ul>	<p>Utiliza de manera eficiente el entorno de desarrollo integrado (IDE) para la elaboración de los programas informáticos.</p> <p>Reporte de la Práctica 1 Programa “Hola Mundo”</p> <p>Reporte de la Práctica 2 Programa “Cálculo de área y perímetro de figura geométrica”</p>	<p>Instrumento de Evaluación:</p> <ul style="list-style-type: none"> <li>Lista de cotejo</li> </ul> <p>Criterios de Forma (estilo):</p> <ul style="list-style-type: none"> <li>Para la Practica 1: Imprimir la cadena de caracteres “Hola Mundo” en pantalla.</li> <li>Para la Practica 2: Imprimir en pantalla el área y perímetro de la figura geométrica solicitada.</li> <li>Para ambas prácticas, resolver problemas reales y cotidianos del entorno.</li> </ul> <p>Criterios de Fondo (parte técnica):</p> <ul style="list-style-type: none"> <li>Para ambas prácticas, ejecutarse correctamente sin errores en tiempo de ejecución, ni de compilación.</li> <li>Para ambas prácticas, el código estará indentado y comentado.</li> </ul>







**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

<b>Unidad Didáctica 2:</b>	Elementos básicos de la programación	<b>Nivel:</b>	3er nivel
<b>Propósito General:</b>	Construye programas informáticos en un lenguaje de programación, para resolver problemas de informática, con base en los algoritmos, en las estructuras de selección, repetición, en los arreglos unidimensionales, multidimensionales, en la ordenación y búsqueda en listas de forma analítica, creativa e innovadora, considerando un enfoque de desarrollo practico y sustentable.		
<b>Unidad de Competencia No 1:</b>	Explica la estructura de un programa y los elementos básicos que lo componen, mediante instrucciones, estructuras de control, sentencias, datos, constantes, variables, expresiones, funciones internas y operaciones de entrada - salida para resolver problemas comunes de informática con un enfoque creativo, de innovación y de trabajo en equipo.		
<b>Aprendizaje Esperado No 1:</b>	Codifica programas utilizando estructuras de control condicionales y selectivas para aumentar su funcionalidad, mediante los tipos primitivos de datos, los operadores aritméticos y la conversión de tipos de datos (casting), construyendo funciones de manera creativa, asertiva y con innovación.	<b>Tiempo estimado para obtener el Aprendizaje Esperado:</b>	9 horas

**Contenidos de Aprendizaje**

Conceptuales	Procedimentales	Actitudinales
<ol style="list-style-type: none"> <li>1. Expresiones               <ol style="list-style-type: none"> <li>1.1. Aritméticas</li> <li>1.2. Lógicas (booleanas)</li> </ol> </li> <li>2. Reglas de prioridad</li> <li>3. Funciones internas</li> <li>4. Conversión de tipo de datos</li> <li>5. El flujo de control de un programa</li> <li>6. Estructura secuencial</li> <li>7. Estructuras selectivas</li> <li>8. Alternativas               <ol style="list-style-type: none"> <li>8.1. Simple (si-entonces/if-then)</li> <li>8.2. Doble (si-entonces-sino/if-then-else)</li> <li>8.3. Múltiple (según_sea, caso de/case)</li> </ol> </li> <li>9. Estructuras de decisión anidadas (en escalera)</li> <li>10. La sentencia ir-a (goto)</li> <li>11. Estructuras</li> <li>12. Repetitivas</li> <li>13. Mientras ("while")               <ol style="list-style-type: none"> <li>13.1. Ejecución de un bucle cero veces</li> <li>13.2. Bucles infinitos</li> <li>13.3. Terminación de bucles con datos de entrada</li> </ol> </li> <li>14. Estructura hacer-mientras ("do-while") .               <ol style="list-style-type: none"> <li>14.1. Diferencias entre mientras (while) y hacer-mientras (do-while)</li> </ol> </li> <li>15. Estructura repetir ("repeat")</li> <li>16. Estructura desde/para ("for")</li> </ol>	<p>Construye programas utilizando operadores, variables, constantes, tipos primitivos de datos, operadores aritméticos y conversión de tipos de datos, para resolver problemas comunes de informática.</p> <p>Practica 3 Programa utilizando "alternativas dobles"</p>	<p>Se interesa en la resolución de problemas comunes de informática de una manera creativa, asertiva e innovadora. Respeto el trabajo de sus compañeros y docentes permitiendo el desarrollo de las actividades de una manera colaborativa.</p>





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

<ul style="list-style-type: none"> <li>16.1. Otras representaciones de estructuras repetitivas desde/para (for)</li> <li>16.2. Realización de una estructura desde con estructura mientras</li> <li>17. Salidas internas de los bucles</li> <li>18. Sentencias de salto interrumpir (break) y continuar (continue)</li> <li>19. Sentencia interrumpir (break)</li> <li>20. Sentencia continuar (continue)</li> <li>21. Comparación de bucles while, for y do-while</li> <li>22. Diseño de bucles (lazos)             <ul style="list-style-type: none"> <li>22.1. Bucles para diseño de sumas y productos</li> <li>22.2. Fin de un bucle</li> </ul> </li> <li>23. Estructuras repetitivas anidadas             <ul style="list-style-type: none"> <li>23.1. Bucles (lazos) anidados</li> </ul> </li> </ul>		
--	--	--

**Estrategia Didáctica y Ambiente de Aprendizaje**

Se propone una metodología **basada en problemas**, para acompañar el proceso de enseñanza-aprendizaje dentro del aula.

**Apertura:** El docente realiza un repaso del uso del entorno de desarrollo integrado, revisa y evalúa las prácticas 1 y 2 para identificar algún tipo de duda que llegarán a tener los alumnos.

**Desarrollo:** El docente en un primer momento, hace una presentación electrónica sobre las expresiones aritméticas y lógicas de un programa, así como las reglas de prioridad, las funciones internas y la conversión de los tipos de datos en un programa. El docente en un segundo momento ejemplifica mediante varios programas diversas problemáticas a ser resueltas incluyendo expresiones aritméticas y lógicas de un programa, reglas de prioridad, funciones internas y la conversión de los tipos de datos en un programa. Los estudiantes buscan, discriminan y sintetizan información para construir sus propias respuestas sobre las diversas problemáticas, el docente supervisa y retroalimenta las aportaciones de los alumnos.

El docente en un tercer momento hace una presentación electrónica sobre las estructuras secuenciales, selectivas, de decisión anidada y de las estructuras repetitivas, que incluyen los bucles, las sentencias de salto y hasta los bucles anidados. El docente en un cuarto momento ejemplifica mediante varios programas diversas problemáticas a ser resueltas incluyendo estructuras secuenciales, selectivas, de decisión anidada y estructuras repetitivas, incluyendo las sentencias de salto, bucles y bucles anidados. Los estudiantes buscan, discriminan y sintetizan información para construir sus propias soluciones a las diversas problemáticas planteadas por el docente. El docente supervisa y retroalimenta las aportaciones de los alumnos.

**Cierre:** El docente resume los aprendizajes de los estudiantes fomentando a la corrección de errores más comunes en la codificación de programas les solicita que realicen de manera individual la evidencia de aprendizaje formativa que es la práctica 3 propuesta.

**Ambiente de Aprendizaje:**

Es un ambiente de aprendizaje presencial pero además el docente creará un aula virtual en alguna plataforma tecnológica como Teams, Classroom, etc. Para poder tener comunicación con los alumnos de manera asíncrona y con ello complementar la comunicación, resolver dudas y crear un ambiente híbrido (presencial y virtual).

Elaboración de Practica 3 Programa utilizando “alternativas dobles”





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

Herramientas Tecnológicas y Recursos Didácticos	Evidencia de Aprendizaje para la Evaluación Formativa	Instrumento y Criterios de Evaluación
<p>Herramientas Tecnológicas:</p> <ul style="list-style-type: none"> <li>• Equipo de Cómputo (PC o Lap Top)</li> <li>• Conexión a internet</li> <li>• Video proyector</li> <li>• Software de Desarrollo de aplicaciones o IDE</li> <li>• Aulas Virtuales</li> </ul> <p>Recursos Didácticos:</p> <ul style="list-style-type: none"> <li>• Libros digitales o impresos relacionados al tema</li> <li>• Infografías y Videos relacionados al tema</li> <li>• Presentaciones electrónicas relacionados al tema</li> <li>• Formularios interactivos relacionados al tema</li> <li>• Páginas o sitios web y aplicaciones móviles</li> </ul>	<p>Reporte de la Practica 3 Programa utilizando “alternativas dobles”</p>	<p>Instrumento de Evaluación:</p> <ul style="list-style-type: none"> <li>• Lista de cotejo</li> </ul> <p>Criterios de Forma (estilo):</p> <ul style="list-style-type: none"> <li>• Imprimir en pantalla la alternativa correcta dependiendo de una acción cuando se cumplen dos determinadas condiciones.</li> <li>• Resolver problemas reales y cotidianos del entorno.</li> </ul> <p>Criterios de Fondo (parte técnica):</p> <ul style="list-style-type: none"> <li>• Ejecutarse correctamente sin errores en tiempo de ejecución, ni de compilación.</li> <li>• El código estará indentado y comentado.</li> </ul>





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

<b>Unidad Didáctica 2:</b>	Elementos básicos de la programación	<b>Nivel:</b>	3er nivel
<b>Propósito General:</b>	Construye programas informáticos en un lenguaje de programación, para resolver problemas de informática, con base en los algoritmos, en las estructuras de selección, repetición, en los arreglos unidimensionales, multidimensionales, en la ordenación y búsqueda en listas de forma analítica, creativa e innovadora, considerando un enfoque de desarrollo practico y sustentable.		
<b>Unidad de Competencia No 1:</b>	Desarrolla programas considerando los elementos básicos que lo componen, mediante operaciones de entrada - salida para resolver problemas comunes de informática con un enfoque creativo, de innovación y de trabajo en equipo.		
<b>Aprendizaje Esperado No 2:</b>	Construye programas que implementen funciones para resolver problemas comunes de informática con un enfoque creativo, de innovación y de trabajo en equipo.	<b>Tiempo estimado para obtener el Aprendizaje Esperado:</b>	9 horas

**Contenidos de Aprendizaje**

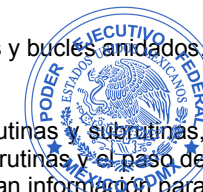
Conceptuales	Procedimentales	Actitudinales
<ol style="list-style-type: none"> <li>1. Funciones               <ol style="list-style-type: none"> <li>1.1. Declaración de funciones</li> <li>1.2. Invocación a las funciones</li> </ol> </li> <li>2. Procedimientos (subrutinas)               <ol style="list-style-type: none"> <li>2.1. Sustitución de argumentos/parámetros</li> </ol> </li> <li>3. Ámbito: variables locales y globales</li> <li>4. Comunicación con subprogramas               <ol style="list-style-type: none"> <li>4.1. Paso de parámetros</li> <li>4.2. Paso por valor</li> <li>4.3. Paso por referencia</li> <li>4.4. Comparaciones de los métodos de paso de parámetros</li> </ol> </li> <li>5. Funciones y procedimientos como parámetros</li> <li>6. Recursión (recursividad)</li> <li>7. Sobrecarga de funciones</li> <li>8.</li> </ol>	<p>Crea funciones para resolver problemas comunes de informática</p> <p>Utiliza procedimientos y subrutinas para la comunicación entre programas y resolver problemas comunes de informática.</p> <p>Realiza programas que utilizan recursividad para resolver problemas comunes de informática</p> <p>Practica 4 Programa utilizando “Funciones”</p>	<p>Se interesa en la resolución de problemas comunes de informática de una manera creativa, asertiva e innovadora.</p> <p>Respeta el trabajo de sus compañeros y docentes permitiendo el desarrollo de las actividades de una manera colaborativa.</p>

**Estrategia Didáctica y Ambiente de Aprendizaje**

Se propone una metodología **basada en problemas**, para acompañar el proceso de enseñanza-aprendizaje dentro del aula.

**Apertura:** El docente realiza un repaso de las estructuras secuenciales, selectivas, de decisión anidada y estructuras repetitivas, incluyendo las sentencias de salto, bucles y bucles anidados, revisa y evalúa la práctica 3 para identificar algún tipo de duda que llegarán a tener los alumnos.

**Desarrollo:** El docente en un primer momento, hace una presentación electrónica sobre las funciones, su declaración e invocación, así como de los procedimientos, rutinas y subrutinas, ejemplifica sobre un programa que va explicando paso a paso. El docente en un segundo momento hace otra presentación electrónica sobre la comunicación entre las subrutinas y el paso de parámetros, la recursividad y la sobrecarga de las funciones, ejemplifica sobre un programa que va explicando paso a paso. Los estudiantes buscan, discriminan y sintetizan información para construir sus propias respuestas sobre las diversas problemáticas, el docente supervisa y retroalimenta las aportaciones de los alumnos.







**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

**Cierre:** El docente resume los aprendizajes de los estudiantes fomentando a la corrección de errores más comunes en la codificación de programas les solicita que realicen de manera individual la evidencia de aprendizaje formativa que es la práctica 4 propuesta.

**Ambiente de Aprendizaje:**

Es un ambiente de aprendizaje presencial pero además el docente creará un aula virtual en alguna plataforma tecnológica como Teams, Classroom, etc. Para poder tener comunicación con los alumnos de manera asíncrona y con ello complementar la comunicación, resolver dudas y crear un ambiente híbrido (presencial y virtual).

Elaboración de Practica 4 Programa utilizando “Funciones”

Herramientas Tecnológicas y Recursos Didácticos	Evidencia de Aprendizaje para la Evaluación Formativa	Instrumento y Criterios de Evaluación
<p>Herramientas Tecnológicas:</p> <ul style="list-style-type: none"> <li>• Equipo de Cómputo (PC o Lap Top)</li> <li>• Conexión a internet</li> <li>• Video proyector</li> <li>• Software de Desarrollo de aplicaciones o IDE</li> <li>• Aulas Virtuales</li> </ul> <p>Recursos Didácticos:</p> <ul style="list-style-type: none"> <li>• Libros digitales o impresos relacionados al tema</li> <li>• Infografías y Videos relacionados al tema</li> <li>• Presentaciones electrónicas relacionados al tema</li> <li>• Formularios interactivos relacionados al tema</li> <li>• Páginas o sitios web y aplicaciones móviles</li> </ul>	<p>Reporte de la Practica 4 Programa utilizando “Funciones”</p>	<p>Instrumento de Evaluación:</p> <ul style="list-style-type: none"> <li>• Lista de cotejo</li> </ul> <p>Criterios de Forma (estilo):</p> <ul style="list-style-type: none"> <li>• Imprimir en pantalla la solución correcta a la función: <math>y = x^n</math> (potencia n de x).</li> <li>• Resolver problemas reales y cotidianos del entorno.</li> </ul> <p>Criterios de Fondo (parte técnica):</p> <ul style="list-style-type: none"> <li>• Ejecutarse correctamente sin errores en tiempo de ejecución, ni de compilación.</li> <li>• El código estará indentado y comentado.</li> </ul>





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

<b>Unidad Didáctica 3:</b>	Arreglos y estructuras de datos	<b>Nivel:</b>	3er Nivel
<b>Propósito General:</b>	Construye programas informáticos en un lenguaje de programación, para resolver problemas de informática, con base en los algoritmos, en las estructuras de selección, repetición, en los arreglos unidimensionales, multidimensionales, en la ordenación y búsqueda en listas de forma analítica, creativa e innovadora, considerando un enfoque de desarrollo práctico y sustentable.		
<b>Unidad de Competencia No 1:</b>	Desarrolla programas utilizando arreglos o arrays mediante la manipulación de estructuras básicas de datos para ordenación y búsqueda, de manera analítica, eficiente y creativa.		
<b>Aprendizaje Esperado No 1:</b>	Construye programas que utilizan arreglos unidimensionales y multidimensionales mediante las estructuras de datos, operación con vectores para solucionar y resolver problemas reales y cotidianos con innovación, adaptabilidad, empatía y proactividad.	<b>Tiempo estimado para obtener el Aprendizaje Esperado:</b>	9 horas

**Contenidos de Aprendizaje**

Conceptuales	Procedimentales	Actitudinales
<ol style="list-style-type: none"> <li>1. Introducción a las estructuras de datos</li> <li>2. Arrays (arreglos) unidimensionales: los vectores</li> <li>3. Operaciones con vectores               <ol style="list-style-type: none"> <li>3.1. Asignación</li> <li>3.2. Lectura/escritura de datos</li> <li>3.3. Acceso secuencial al vector (recorrido)</li> <li>3.4. Actualización de un vector</li> </ol> </li> <li>4. Arrays de varias dimensiones               <ol style="list-style-type: none"> <li>4.1. Arrays bidimensionales (tablas/matrices)</li> </ol> </li> <li>5. Arrays multidimensionales</li> <li>6. Almacenamiento de arrays en memoria               <ol style="list-style-type: none"> <li>6.1. Almacenamiento de un vector</li> <li>6.2. Almacenamiento de arrays multidimensionales</li> </ol> </li> </ol>	<p>Utiliza las estructuras de datos, operación con vectores para realizar ordenación y búsqueda de elementos.</p> <p>Aplica arrays unidimensionales y multidimensionales para resolver problemas reales y cotidianos.</p> <p>Practica 5 Programa utilizando “Arreglos”</p>	<p>Respetar las ideas y opiniones de sus compañeros y docentes. Se interesa por la resolución de los problemas encontrados en su entorno con innovación, adaptabilidad, empatía y proactividad.</p> <p>Atiende las problemáticas de su entorno, resolviendo con trabajo colaborativo.</p>

**Estrategia Didáctica y Ambiente de Aprendizaje**

Se propone una metodología **basada en problemas**, para acompañar el proceso de enseñanza-aprendizaje dentro del aula.

**Apertura:** El docente realiza un repaso de las funciones, su declaración e invocación, de los procedimientos, las rutinas, subrutinas, la comunicación entre las subrutinas, el paso de parámetros, la recursividad y la sobrecarga de las funciones.

**Desarrollo:** El docente hace una presentación electrónica sobre los conceptos básicos de las estructuras de datos básicas, arreglos (arrays) unidimensionales, multidimensionales y del almacenamiento en memoria, ejemplifica sobre un programa que va explicando paso a paso. Los estudiantes buscan, discriminan y sintetizan información para construir sus propias respuestas sobre las diversas problemáticas, el docente supervisa y retroalimenta las aportaciones de los alumnos.





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

**Cierre:** El docente resume los aprendizajes de los estudiantes fomentando a la corrección de errores más comunes en la codificación de programas les solicita que realicen de manera individual la evidencia de aprendizaje formativa que es la práctica 5 propuesta.

**Ambiente de Aprendizaje:**

Es un ambiente de aprendizaje presencial pero además el docente creará un aula virtual en alguna plataforma tecnológica como Teams, Classroom, etc. Para poder tener comunicación con los alumnos de manera asíncrona y con ello complementar la comunicación, resolver dudas y crear un ambiente híbrido (presencial y virtual).

Elaboración de Practica 5 Programa utilizando “Arreglos”

Herramientas Tecnológicas y Recursos Didácticos	Evidencia de Aprendizaje para la Evaluación Formativa	Instrumento y Criterios de Evaluación
<p>Herramientas Tecnológicas:</p> <ul style="list-style-type: none"> <li>• Equipo de Cómputo (PC o Lap Top)</li> <li>• Conexión a internet</li> <li>• Video proyector</li> <li>• Software de Desarrollo de aplicaciones o IDE</li> <li>• Aulas Virtuales</li> </ul> <p>Recursos Didácticos:</p> <ul style="list-style-type: none"> <li>• Libros digitales o impresos relacionados al tema</li> <li>• Infografías y Videos relacionados al tema</li> <li>• Presentaciones electrónicas relacionados al tema</li> <li>• Formularios interactivos relacionados al tema</li> <li>• Páginas o sitios web y aplicaciones móviles</li> </ul>	<p>Reporte de la Practica 5 Programa utilizando “Arreglos”</p>	<p>Instrumento de Evaluación:</p> <ul style="list-style-type: none"> <li>• Lista de cotejo</li> </ul> <p>Criterios de Forma (estilo):</p> <ul style="list-style-type: none"> <li>• Imprimir en pantalla un arreglo multidimensional de forma correcta.</li> <li>• Resolver problemas reales y cotidianos del entorno.</li> </ul> <p>Criterios de Fondo (parte técnica):</p> <ul style="list-style-type: none"> <li>• Ejecutarse correctamente sin errores en tiempo de ejecución, ni de compilación.</li> <li>• El código estará indentado y comentado.</li> </ul>





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

<b>Unidad Didáctica 3:</b>	Arreglos y estructuras de datos	<b>Nivel:</b>	3er nivel
<b>Propósito General:</b>	Construye programas informáticos en un lenguaje de programación, para resolver problemas de informática, con base en los algoritmos, en las estructuras de selección, repetición, en los arreglos unidimensionales, multidimensionales, en la ordenación y búsqueda en listas de forma analítica, creativa e innovadora, considerando un enfoque de desarrollo práctico y sustentable.		
<b>Unidad de Competencia No 1:</b>	Desarrolla programas utilizando arreglos o arrays mediante la manipulación de estructuras básicas de datos para ordenación y búsqueda, de manera analítica, eficiente y creativa.		
<b>Aprendizaje Esperado No 2:</b>	Construye programas que utilizan uniones, enumeraciones, ordenación de listas y búsquedas secuenciales para solucionar y resolver problemas reales y cotidianos, con innovación, adaptabilidad, empatía y proactividad.	<b>Tiempo estimado para obtener el Aprendizaje Esperado:</b>	9 horas

**Contenidos de Aprendizaje**

Conceptuales	Procedimentales	Actitudinales
<ol style="list-style-type: none"> <li>1. Estructuras versus registros               <ol style="list-style-type: none"> <li>1.1. Registros</li> </ol> </li> <li>2. Arrays de estructuras</li> <li>3. Uniones</li> <li>4. Unión versus estructura</li> <li>5. Enumeraciones</li> <li>6. Ordenación               <ol style="list-style-type: none"> <li>6.1. Método de intercambio o de burbuja</li> <li>6.2. Ordenación por inserción</li> <li>6.3. Ordenación por selección</li> <li>6.4. Método de Shell</li> <li>6.5. Método de ordenación rápida (quicksort)</li> </ol> </li> <li>7. Búsqueda               <ol style="list-style-type: none"> <li>7.1. Búsqueda secuencial</li> <li>7.2. Búsqueda binaria</li> <li>7.3. Búsqueda mediante transformación de claves (hasting)</li> </ol> </li> <li>8. Intercalación</li> </ol>	<p>Utiliza uniones versus estructuras para la resolución de problemas informáticos comunes.</p> <p>Realiza programas que utilizan la ordenación y las búsquedas para resolver problemas reales y cotidianos de su entorno.</p> <p>Practica 6 Programa utilizando “Ordenación”</p>	<p>Respeto las ideas y opiniones de sus compañeros y docentes.</p> <p>Se interesa por la resolución de los problemas encontrados en su entorno con innovación, adaptabilidad, empatía y proactividad.</p> <p>Atiende las problemáticas de su entorno, trabajando con una comunicación asertiva y de forma empática.</p>

**Estrategia Didáctica y Ambiente de Aprendizaje**

Se propone una metodología **basada en problemas**, para acompañar el proceso de enseñanza-aprendizaje dentro del aula.

**Apertura:** El docente realiza un repaso de las estructuras de datos básicas, arreglos (arrays) unidimensionales, multidimensionales y del almacenamiento en memoria.

**Desarrollo:** El docente hace una presentación electrónica sobre los conceptos de otras estructuras de datos tales como los registros, uniones, enumeraciones, ordenación y búsqueda, ejemplifica sobre un programa que va explicando paso a paso. Los estudiantes buscan, discriminan y sintetizan información para construir sus propias respuestas sobre las diversas problemáticas, el docente supervisa y retroalimenta las aportaciones de los alumnos.







**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

**Cierre:** El docente resume los aprendizajes de los estudiantes fomentando a la corrección de errores más comunes en la codificación de programas les solicita que realicen de manera individual la evidencia de aprendizaje formativa que es la práctica 6 propuesta.

**Ambiente de Aprendizaje:**

Es un ambiente de aprendizaje presencial pero además el docente creará un aula virtual en alguna plataforma tecnológica como Teams, Classroom, etc. Para poder tener comunicación con los alumnos de manera asíncrona y con ello complementar la comunicación, resolver dudas y crear un ambiente híbrido (presencial y virtual).

Elaboración de Practica 6 Programa utilizando "Ordenación"

Herramientas Tecnológicas y Recursos Didácticos	Evidencia de Aprendizaje para la Evaluación Formativa	Instrumento y Criterios de Evaluación
<p>Herramientas Tecnológicas:</p> <ul style="list-style-type: none"> <li>• Equipo de Cómputo (PC o Lap Top)</li> <li>• Conexión a internet</li> <li>• Video proyector</li> <li>• Software de Desarrollo de aplicaciones o IDE</li> <li>• Aulas Virtuales</li> </ul> <p>Recursos Didácticos:</p> <ul style="list-style-type: none"> <li>• Libros digitales o impresos relacionados al tema</li> <li>• Infografías y Videos relacionados al tema</li> <li>• Presentaciones electrónicas relacionados al tema</li> <li>• Formularios interactivos relacionados al tema</li> <li>• Páginas o sitios web y aplicaciones móviles</li> </ul>	<p>Reporte de la Practica 6 Programa utilizando "Ordenación"</p>	<p>Instrumento de Evaluación:</p> <ul style="list-style-type: none"> <li>• Lista de cotejo</li> </ul> <p>Criterios de Forma (estilo):</p> <ul style="list-style-type: none"> <li>• Imprimir en pantalla la ordenación de un arreglo mediante el método de la burbuja de forma correcta.</li> <li>• Resolver problemas reales y cotidianos del entorno.</li> </ul> <p>Criterios de Fondo (parte técnica):</p> <ul style="list-style-type: none"> <li>• Ejecutarse correctamente sin errores en tiempo de ejecución, ni de compilación.</li> <li>• El código estará indentado y comentado.</li> </ul>



**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

**PRÁCTICAS**

<b>Nombre de la Práctica:</b>	Práctica 1 Programa “Hola Mundo”	<b>N° de la Práctica:</b>	1	<b>Tiempo:</b>	6 horas
<b>Unidad de Competencia:</b>	Relaciona los conceptos y paradigmas de programación, utilizando herramientas para el aprendizaje de los procesos de compilación y ejecución de los programas con base en un entorno de desarrollo integrado (IDE) de forma innovadora, eficiente y creativa.				
<b>Aprendizajes Esperados Relacionados con la Práctica:</b>	Codifica programas mediante un editor y compilador independientes y en un entorno de desarrollo integrado (IDE) para la resolución de problemas diversos con un enfoque de innovación, eficiente y creativo				
Contenidos de Aprendizaje Relacionados con la Práctica					
Conceptuales		Procedimentales		Actitudinales	
1. El entorno de desarrollo integrado 1.1. IDE Netbeans o similar		Utiliza de manera eficiente el entorno de desarrollo integrado (IDE) para la elaboración de los programas informáticos.		Atiende una actitud respetuosa hacia sus compañeros y docentes, el desarrollo de sus actividades las realiza de una forma analítica, reflexiva, con una comunicación asertiva y el trabajo se cumple de forma colaborativa.	
Estrategia Didáctica y Ambiente de Aprendizaje					
<p><b>Estrategia Didáctica:</b> El docente presenta como se descarga e instala el entorno de desarrollo integrado, muestra instalación, configuración y utilización del entorno de desarrollo integrado para que los estudiantes sigan los procedimientos de mostrados del mismo IDE. El docente, con el apoyo de un video proyector y una pc o equipo portátil, comienza a codificar el programa ejemplo y solicita a los estudiantes que abran su IDE y que junto con él continúen codificando el programa ejemplo. Una vez terminado el programa ejemplo, solicita a los estudiantes que inicien la codificación de su práctica 1 programa “Hola Mundo” todo esto basado en el ejemplo que se les mostró en pantalla. Los estudiantes codifican su practica 1 programa “Hola Mundo”, efectúan la prueba de funcionalidad validando que la práctica este sin errores en tiempo de ejecución, ni de compilación, que el código se encuentre indentado y comentado empleando los conceptos básicos de la programación, atendiendo una actitud respetuosa hacia sus compañeros y docentes, el desarrollo de sus actividades las realiza de una forma analítica, reflexiva, con una comunicación efectiva y el trabajo se cumple de forma colaborativa, finalmente lo entregan al docente en el aula virtual.</p> <p><b>Ambiente de Aprendizaje:</b> Laboratorios de computación y en el aula virtual o plataforma tecnológica como Teams, Classroom, etc. para con ello complementar la comunicación con los alumnos y crear un ambiente híbrido presencial y virtual</p>					
Herramientas Tecnológicas y Recursos Didácticos		Evidencia de Aprendizaje para la Evaluación Formativa		Criterios e Instrumentos de Evaluación	
Herramientas Tecnológicas: <ul style="list-style-type: none"> <li>Equipo de Cómputo (PC o Lap Top)</li> <li>Software de Desarrollo de aplicaciones o IDE</li> <li>Video proyector</li> <li>Aulas Virtuales</li> </ul> Recursos Didácticos: <ul style="list-style-type: none"> <li>Presentaciones electrónicas relacionados al tema</li> <li>Páginas o sitios web y relacionados al tema</li> </ul>		Reporte de la Práctica 1 Programa “Hola Mundo”		Instrumento de Evaluación: <ul style="list-style-type: none"> <li>Lista de cotejo</li> </ul> Criterios de Forma (estilo): <ul style="list-style-type: none"> <li>Imprimir la cadena de caracteres “Hola Mundo” en pantalla.</li> <li>Resolver problemas reales y cotidianos del entorno.</li> </ul> Criterios de Fondo (parte técnica): <ul style="list-style-type: none"> <li>Ejecutarse correctamente sin errores en tiempo de ejecución, ni de compilación.</li> <li>El código estará indentado y comentado.</li> </ul>	





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

<b>Nombre de la Práctica:</b>	Práctica 2 Programa “Cálculo de área y perímetro de figura geométrica	<b>Nº de la Práctica:</b>	2	<b>Tiempo:</b>	6 horas
<b>Unidad de Competencia:</b>	Relaciona los conceptos y paradigmas de programación, utilizando herramientas para el aprendizaje de los procesos de compilación y ejecución de los programas con base en un entorno de desarrollo integrado (IDE) de forma innovadora, eficiente y creativa.				
<b>Aprendizajes Esperados Relacionados con la Práctica:</b>	Codifica programas de cálculo de áreas y perímetros mediante la estructura de programas, la asignación y declaración de variables en un entorno de desarrollo integrado (IDE) para la resolución de problemas diversos con un enfoque de innovación, eficiente y creativo				
Contenidos de Aprendizaje Relacionados con la Práctica					
Conceptuales		Procedimentales		Actitudinales	
2. La operación de asignación 2.1. Aritmética 2.2. Lógica 2.3. De cadenas de caracteres 2.4. Múltiple 3. Estructura de programas 3.1. Cabecera del programa 3.2. Declaración de variables		Utiliza de manera eficiente el entorno de desarrollo integrado (IDE) para la elaboración de los programas informáticos.		Atiende una actitud respetuosa hacia sus compañeros y docentes, el desarrollo de sus actividades las realiza de una forma analítica, reflexiva, con una comunicación efectiva y el trabajo se cumple de forma colaborativa.	
Estrategia Didáctica y Ambiente de Aprendizaje					
<p><b>Estrategia Didáctica:</b>                  El docente presenta una figura geométrica, así como también las fórmulas para el cálculo del área y perímetro de esta figura, para que los estudiantes recuerden dichas formulas. El docente, con el apoyo de un video proyector y una pc o equipo portátil, comienza a codificar la resolución a la problemática del cálculo del área y perímetro de la figura mostrada en el ejemplo. Solicita a los estudiantes que abran su IDE y que junto con él continúen codificando la resolución de la problemática ejemplo. Una vez terminado el programa ejemplo, solicita a los estudiantes que inicien la codificación de su práctica 2 Programa “Cálculo de área y perímetro de figura geométrica” todo esto basado en el ejemplo que se les mostró en pantalla.                  Los estudiantes codifican su práctica 2 Programa “Cálculo de área y perímetro de figura geométrica”, efectúan la prueba de funcionalidad validando que la práctica este sin errores en tiempo de ejecución, ni de compilación, que el código se encuentre indentado y comentado empleando los conceptos básicos de la programación, atendiendo una actitud respetuosa hacia sus compañeros y docentes, el desarrollo de sus actividades las realiza de una forma analítica, reflexiva, con una comunicación efectiva y el trabajo se cumple de forma colaborativa, finalmente lo entregan al docente en el aula virtual.</p> <p><b>Ambiente de Aprendizaje:</b>                  Laboratorios de computación y en el aula virtual o plataforma tecnológica como Teams, Classroom, etc. para con ello complementar la comunicación con los alumnos y crear un ambiente híbrido presencial y virtual</p>					
Herramientas Tecnológicas y Recursos Didácticos		Evidencia de Aprendizaje para la Evaluación Formativa		Criterios e Instrumentos de Evaluación	
Herramientas Tecnológicas: <ul style="list-style-type: none"> <li>Equipo de Cómputo (PC o Lap Top)</li> <li>Software de Desarrollo de aplicaciones o IDE</li> <li>Video proyector</li> <li>Aulas Virtuales</li> </ul> Recursos Didácticos: <ul style="list-style-type: none"> <li>Libros digitales o impresos relacionados al tema</li> <li>Infografías y Videos relacionados al tema</li> </ul>		Reporte de la Práctica 2 Programa “Cálculo de área y perímetro de figura geométrica”		Instrumento de Evaluación: <ul style="list-style-type: none"> <li>Lista de cotejo</li> </ul> Criterios de Forma (estilo): <ul style="list-style-type: none"> <li>Imprimir en pantalla el área y perímetro de la figura geométrica solicitada.</li> <li>Resolver problemas reales y cotidianos del entorno.</li> </ul> Criterios de Fondo (parte técnica): <ul style="list-style-type: none"> <li>Ejecutarse correctamente sin errores en tiempo de ejecución, ni de compilación.</li> <li>El código estará indentado y comentado.</li> </ul>	



**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

<b>Nombre de la Práctica:</b>	Practica 3 Programa utilizando “alternativas dobles”	<b>N° de la Práctica:</b>	3	<b>Tiempo:</b>	6 horas
<b>Unidad de Competencia:</b>	Desarrolla programas considerando los elementos básicos que lo componen, mediante operaciones de entrada - salida para resolver problemas comunes de informática con un enfoque creativo, de innovación y de trabajo en equipo.				
<b>Aprendizajes Esperados Relacionados con la Práctica:</b>	Codifica programas utilizando la estructura de control selectiva doble para aumentar su funcionalidad, resolviendo problemas de la vida cotidiana de manera creativa, asertiva y con innovación				

**Contenidos de Aprendizaje Relacionados con la Práctica**

Conceptuales	Procedimentales	Actitudinales
7. Estructuras selectivas 8. Alternativas 8.1. Simple (si-entonces/if-then) 8.2. Doble (si-entonces-sino/if-then-else)	Construye programas utilizando la estructura selectiva doble para resolver problemas comunes de informática.	Se interesa en la resolución de problemas comunes de informática de una manera creativa, asertiva e innovadora. Respeto el trabajo de sus compañeros y docentes permitiendo el desarrollo de las actividades de una manera colaborativa.

**Estrategia Didáctica y Ambiente de Aprendizaje**

**Estrategia Didáctica:**

El docente presenta una problemática en donde se ejecuta una acción cuando se cumple dos determinadas condiciones. La estructura selectiva **si-entonces-sino** evalúa la condición y si la condición es verdadera, entonces ejecuta la acción 1 (en esta acción se incluye otra selección si-entonces y dependiendo del resultado o acciones caso de ser 1 una acción compuesta y constar de varias acciones) si la condición es falsa, entonces no hacer nada, para que los estudiantes comprendan y analicen la problemática. El docente, con el apoyo de un video proyector y una pc o equipo portátil, comienza a codificar la resolución a la problemática de una estructura selectiva si-entonces-sino mostrada en el ejemplo. Solicita a los estudiantes que abran su IDE y que junto con él continúen codificando la resolución de la problemática ejemplo. Una vez terminado el programa ejemplo, solicita a los estudiantes que inicien la codificación de su práctica 3 Programa utilizando “alternativas dobles” todo esto basado en el ejemplo que se les mostró en pantalla.

Los estudiantes codifican su práctica 3 Programa utilizando “alternativas dobles”, efectúan la prueba de funcionalidad validando que la práctica este sin errores en tiempo de ejecución, ni de compilación, que el código se encuentre indentado y comentado empleando la estructura con alternativa doble, atendiendo un interés en la resolución de problemas comunes de informática de una manera creativa, asertiva e innovadora. Respetando el trabajo permitiendo el desarrollo de las actividades de una manera colaborativa y respetuosa hacia sus compañeros y docentes, finalmente entregan la práctica al docente en el aula virtual.

**Ambiente de Aprendizaje:**

Laboratorios de computación y en el aula virtual o plataforma tecnológica como Teams, Classroom, etc. para con ello complementar la comunicación con los alumnos y crear un ambiente híbrido presencial y virtual

Herramientas Tecnológicas y Recursos Didácticos	Evidencia de Aprendizaje para la Evaluación Formativa	Criterios e Instrumentos de Evaluación
Herramientas Tecnológicas: <ul style="list-style-type: none"> <li>Equipo de Cómputo (PC o Lap Top)</li> <li>Video proyector</li> <li>Software de Desarrollo de aplicaciones o IDE</li> <li>Aulas Virtuales</li> </ul> Recursos Didácticos: <ul style="list-style-type: none"> <li>Libros digitales o impresos relacionados al tema</li> <li>Presentaciones electrónicas relacionados al tema</li> <li>Formularios interactivos relacionados al tema</li> </ul>	Reporte de la Practica 3 Programa utilizando “alternativas dobles”	Instrumento de Evaluación: <ul style="list-style-type: none"> <li>Lista de cotejo</li> </ul> Criterios de Forma (estilo): <ul style="list-style-type: none"> <li>Imprimir en pantalla la alternativa correcta dependiendo de una acción cuando se cumplen dos determinadas condiciones.</li> <li>Resolver problemas reales y cotidianos del entorno.</li> </ul> Criterios de Fondo (parte técnica): <ul style="list-style-type: none"> <li>Ejecutarse correctamente sin errores en tiempo de ejecución, ni de compilación.</li> <li>El código estará indentado y comentado.</li> </ul>



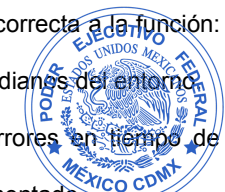




**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

<b>Nombre de la Práctica:</b>	Practica 4 Programa utilizando “Funciones”	<b>N° de la Práctica:</b>	4	<b>Tiempo:</b>	6 horas
<b>Unidad de Competencia:</b>	Desarrolla programas considerando los elementos básicos que lo componen, mediante operaciones de entrada - salida para resolver problemas comunes de informática con un enfoque creativo, de innovación y de trabajo en equipo.				
<b>Aprendizajes Esperados Relacionados con la Práctica:</b>	Construye programas que implementen funciones para resolver problemas comunes de informática con un enfoque creativo, de innovación y de trabajo en equipo				
Contenidos de Aprendizaje Relacionados con la Práctica					
Conceptuales		Procedimentales		Actitudinales	
1. Funciones 1.1. Declaración de funciones 1.2. Invocación a las funciones		Crea funciones para resolver problemas comunes de informática		Se interesa en la resolución de problemas comunes de informática de una manera creativa, asertiva e innovadora. Respeta el trabajo de sus compañeros y docentes permitiendo el desarrollo de las actividades de una manera colaborativa.	
Estrategia Didáctica y Ambiente de Aprendizaje					
<b>Estrategia Didáctica:</b>					
El docente presenta alguna función potencia de base real y exponente natural con el apoyo de un video proyector y una pc o equipo portátil, comienza a codificar la resolución a la función mostrada. Solicita a los estudiantes que abran su IDE y que junto con él continúen codificando la resolución de la problemática ejemplo. Una vez terminado el programa ejemplo, solicita a los estudiantes que inicien la codificación de su practica 4 Programa utilizando “Funciones” todo esto basado en el ejemplo que se les mostró en pantalla.					
Los estudiantes codifican su practica 4 Programa utilizando “Funciones”, efectúan la prueba de funcionalidad validando que la práctica este sin errores en tiempo de ejecución, ni de compilación, que el código se encuentre indentado y comentado empleando la estructura con alternativa doble, atendiendo un interés en la resolución de problemas comunes de informática de una manera creativa, asertiva e innovadora. Respetando el trabajo permitiendo el desarrollo de las actividades de una manera colaborativa y respetuosa hacia sus compañeros y docentes, finalmente entregan la práctica al docente en el aula virtual.					
<b>Ambiente de Aprendizaje:</b>					
Laboratorios de computación y en el aula virtual o plataforma tecnológica como Teams, Classroom, etc. para con ello complementar la comunicación con los alumnos y crear un ambiente híbrido presencial y virtual					
Herramientas Tecnológicas y Recursos Didácticos		Evidencia de Aprendizaje para la Evaluación Formativa		Criterios e Instrumentos de Evaluación	
Herramientas Tecnológicas: <ul style="list-style-type: none"> <li>Equipo de Cómputo (PC o Lap Top)</li> <li>Video proyector</li> <li>Software de Desarrollo de aplicaciones o IDE</li> <li>Aulas Virtuales</li> </ul> Recursos Didácticos: <ul style="list-style-type: none"> <li>Libros digitales o impresos relacionados al tema</li> <li>Infografías y Videos relacionados al tema</li> <li>Presentaciones electrónicas relacionados al tema</li> </ul>		Reporte de la Practica 4 Programa utilizando “Funciones”		Instrumento de Evaluación: <ul style="list-style-type: none"> <li>Lista de cotejo</li> </ul> Criterios de Forma (estilo): <ul style="list-style-type: none"> <li>Imprimir en pantalla la solución correcta a la función: <math>y = x^n</math> (potencia n de x).</li> <li>Resolver problemas reales y cotidianos del entorno.</li> </ul> Criterios de Fondo (parte técnica): <ul style="list-style-type: none"> <li>Ejecutarse correctamente sin errores en tiempo de ejecución, ni de compilación.</li> <li>El código estará indentado y comentado.</li> </ul>	





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

<b>Nombre de la Práctica:</b>	Practica 5 Programa utilizando “Arreglos”	<b>N° de la Práctica:</b>	5	<b>Tiempo:</b>	6 horas
<b>Unidad de Competencia:</b>	Desarrolla programas utilizando arreglos o arrays mediante la manipulación de estructuras básicas de datos para ordenación y búsqueda, de manera analítica, eficiente y creativa.				
<b>Aprendizajes Esperados Relacionados con la Práctica:</b>	Construye programas que utilizan arreglos unidimensionales para solucionar y resolver problemas reales y cotidianos con innovación, adaptabilidad, empatía y proactividad				
Contenidos de Aprendizaje Relacionados con la Práctica					
Conceptuales	Procedimentales	Actitudinales			
3. Operaciones con vectores 3.1. Asignación 3.2. Lectura/escritura de datos 3.3. Acceso secuencial al vector (recorrido) 3.4. Actualización de un vector	Aplica arrays unidimensionales para resolver problemas reales y cotidianos.	Respeta las ideas y opiniones de sus compañeros y docentes. Se interesa por la resolución de los problemas encontrados en su entorno con innovación, adaptabilidad, empatía y proactividad. Atiende las problemáticas de su entorno, resolviendo con trabajo colaborativo.			
Estrategia Didáctica y Ambiente de Aprendizaje					
<b>Estrategia Didáctica:</b> El docente presenta algunos ejemplos de operaciones con arreglos con el apoyo de un video proyector y una pc o equipo portátil, comienza a codificar la resolución a un problema que utiliza arreglos. Solicita a los estudiantes que abran su IDE y que junto con él continúen codificando la resolución de la problemática ejemplo. Una vez terminado el programa ejemplo, solicita a los estudiantes que inicien la codificación de su practica 5 Programa utilizando “Arreglos” todo esto basado en el ejemplo que se les mostró en pantalla. Los estudiantes codifican su practica 5 Programa utilizando “Arreglos”, efectúan la prueba de funcionalidad validando que la práctica este sin errores en tiempo de ejecución, ni de compilación, que el código se encuentre indentado y comentado empleando la estructura con alternativa doble, atendiendo un interés en la resolución de problemas comunes de informática de una manera creativa, asertiva e innovadora. Respetando el trabajo permitiendo el desarrollo de las actividades de una manera colaborativa y respetuosa hacia sus compañeros y docentes, atiende las problemáticas de su entorno, resolviendo con trabajo colaborativo finalmente entregan la práctica al docente en el aula virtual.					
<b>Ambiente de Aprendizaje:</b> Laboratorios de computación y en el aula virtual o plataforma tecnológica como Teams, Classroom, etc. para con ello complementar la comunicación con los alumnos y crear un ambiente híbrido presencial y virtual					
Herramientas Tecnológicas y Recursos Didácticos	Evidencia de Aprendizaje para la Evaluación Formativa	Criterios e Instrumentos de Evaluación			
<b>Herramientas Tecnológicas:</b> <ul style="list-style-type: none"> <li>Equipo de Cómputo (PC o Lap Top)</li> <li>Conexión a internet</li> <li>Video proyector</li> <li>Software de Desarrollo de aplicaciones o IDE</li> <li>Aulas Virtuales</li> </ul> <b>Recursos Didácticos:</b> <ul style="list-style-type: none"> <li>Libros digitales o impresos relacionados al tema</li> <li>Infografías y Videos relacionados al tema</li> <li>Presentaciones electrónicas relacionados al tema</li> <li>Formularios interactivos relacionados al tema</li> <li>Páginas o sitios web y aplicaciones móviles</li> </ul>	Reporte de la Practica 5 Programa utilizando “Arreglos”	<b>Instrumento de Evaluación:</b> <ul style="list-style-type: none"> <li>Lista de cotejo</li> </ul> <b>Criterios de Forma (estilo):</b> <ul style="list-style-type: none"> <li>Imprimir en pantalla un arreglo multidimensional de forma correcta.</li> <li>Resolver problemas reales y cotidianos del entorno.</li> </ul> <b>Criterios de Fondo (parte técnica):</b> <ul style="list-style-type: none"> <li>Ejecutarse correctamente sin errores en tiempo de ejecución, ni de compilación.</li> <li>El código estará indentado y comentado.</li> </ul>			





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

<b>Nombre de la Práctica:</b>	Practica 6 Programa utilizando “Ordenación”	<b>N° de la Práctica:</b>	6	<b>Tiempo:</b>	6 horas
<b>Unidad de Competencia:</b>	Desarrolla programas utilizando arreglos o arrays mediante la manipulación de estructuras básicas de datos para ordenación y búsqueda, de manera analítica, eficiente y creativa.				
<b>Aprendizajes Esperados Relacionados con la Práctica:</b>	Construye programas que utilizan ordenación para solucionar y resolver problemas reales y cotidianos con innovación, adaptabilidad, empatía y proactividad				

**Contenidos de Aprendizaje Relacionados con la Práctica**

Conceptuales	Procedimentales	Actitudinales
6. Ordenación 6.1. Método de intercambio o de burbuja	Realiza programas que utilizan la ordenación y las búsquedas para resolver problemas reales y cotidianos de su entorno.	Respeto las ideas y opiniones de sus compañeros y docentes. Se interesa por la resolución de los problemas encontrados en su entorno con innovación, adaptabilidad, empatía y proactividad. Atiende las problemáticas de su entorno, trabajando con una comunicación asertiva y de forma empática.

**Estrategia Didáctica y Ambiente de Aprendizaje**

**Estrategia Didáctica:**

El docente presenta algunos ejemplos de ordenación y búsqueda de arreglos con el apoyo de un video proyector y una pc o equipo portátil, comienza a codificar la resolución a un problema en donde se tiene que usar la ordenación y búsqueda. Solicita a los estudiantes que abran su IDE y que junto con él continúen codificando la resolución de la problemática ejemplo. Una vez terminado el programa ejemplo, solicita a los estudiantes que inicien la codificación de su practica 6 Programa utilizando “Ordenación” todo esto basado en el ejemplo que se les mostró en pantalla. Los estudiantes codifican su practica 6 Programa utilizando “Ordenación”, efectúan la prueba de funcionalidad validando que la práctica este sin errores en tiempo de ejecución, ni de compilación, que el código se encuentre indentado y comentado empleando el método de intercambio o método de burbuja, atendiendo un interés en la resolución de problemas comunes de informática de una manera creativa, asertiva e innovadora. Respetando el trabajo permitiendo el desarrollo de las actividades de una manera colaborativa y respetuosa hacia sus compañeros y docentes, atiende las problemáticas de su entorno, resolviendo con trabajo colaborativo finalmente entregan la práctica al docente en el aula virtual.

**Ambiente de Aprendizaje:**

Laboratorios de computación y en el aula virtual o plataforma tecnológica como Teams, Classroom, etc. para con ello complementar la comunicación con los alumnos y crear un ambiente hibrido presencial y virtual

Herramientas Tecnológicas y Recursos Didácticos	Evidencia de Aprendizaje para la Evaluación Formativa	Criterios e Instrumentos de Evaluación
<ul style="list-style-type: none"> <li>Equipo de Cómputo (PC o Lap Top)</li> <li>Conexión a internet</li> <li>Video proyector</li> <li>Software de Desarrollo de aplicaciones o IDE</li> <li>Aulas Virtuales</li> </ul> <p>Recursos Didácticos:</p> <ul style="list-style-type: none"> <li>Libros digitales o impresos relacionados al tema</li> <li>Infografías y Videos relacionados al tema</li> <li>Presentaciones electrónicas relacionados al tema</li> <li>Formularios interactivos relacionados al tema</li> <li>Páginas o sitios web y aplicaciones móviles</li> </ul>	Reporte de la Practica 6 Programa utilizando “Ordenación”	<p>Instrumento de Evaluación:</p> <ul style="list-style-type: none"> <li>Lista de cotejo</li> </ul> <p>Criterios de Forma (estilo):</p> <ul style="list-style-type: none"> <li>Imprimir en pantalla la ordenación de un arreglo mediante el método de la burbuja de forma correcta.</li> <li>Resolver problemas reales y cotidianos del entorno.</li> </ul> <p>Criterios de Fondo (parte técnica):</p> <ul style="list-style-type: none"> <li>Ejecutarse correctamente sin errores en tiempo de ejecución, ni de compilación.</li> <li>El código estará indentado y comentado.</li> </ul>





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

**PLAN DE EVALUACIÓN SUMATIVA**

N°	Unidad de Competencia	Evidencia Integradora	Criterios e Instrumento de Evaluación	Porcentaje de Acreditación
1	Relaciona los conceptos y paradigmas de programación, utilizando herramientas para el aprendizaje de los procesos de compilación y ejecución de los programas con base en un entorno de desarrollo integrado (IDE) de forma innovadora, eficiente y creativa	<p>Elaboración de organizador grafico de los conceptos básicos y paradigmas de la programación.</p> <p>Reporte de la Práctica 1 Programa “Hola Mundo”</p> <p>Reporte de la Práctica 2 Programa “Cálculo de área y perímetro de figura geométrica”</p>	<p>Instrumento de Evaluación:</p> <ul style="list-style-type: none"> <li>• Lista de cotejo</li> </ul> <p>Criterios de Forma (estilo):</p> <ul style="list-style-type: none"> <li>• Para el organizador gráfico entrega de trabajo de investigación sin faltas de ortografía</li> <li>• Para el organizador gráfico entrega de documento pdf en el aula y/o plataforma virtual</li> <li>• Para el organizador gráfico entrega de actividad en tiempo y forma solicitados</li> <li>• Para la Practica 1: Imprimir la cadena de caracteres “Hola Mundo” en pantalla.</li> <li>• Para la Practica 2: Imprimir en pantalla el área y perímetro de la figura geométrica solicitada.</li> <li>• Para ambas prácticas, resolver problemas reales y cotidianos del entorno.</li> </ul> <p>Criterios de Fondo (parte técnica):</p> <ul style="list-style-type: none"> <li>• Para el organizador gráfico incluye los conceptos básicos de la programación.</li> <li>• Para el organizador gráfico incluye los tipos de datos y las operaciones primitivas.</li> <li>• Para el organizador gráfico incluye las características más importantes de los paradigmas de la programación.</li> <li>• Para ambas prácticas, ejecutarse correctamente sin errores en tiempo de ejecución, ni de compilación.</li> </ul>	30 %








**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

2	<p>Desarrolla programas considerando los elementos básicos que lo componen, mediante operaciones de entrada - salida para resolver problemas comunes de informática con un enfoque creativo, de innovación y de trabajo en equipo.</p>	<p>Reporte de la Practica 3 Programa utilizando “alternativas dobles”</p> <p>Reporte de la Practica 4 Programa utilizando “Funciones”</p>	<ul style="list-style-type: none"> <li>• Para ambas prácticas, el código estará indentado y comentado.</li> </ul> <p>Instrumento de Evaluación:</p> <ul style="list-style-type: none"> <li>• Lista de cotejo</li> </ul> <p>Criterios de Forma (estilo):</p> <ul style="list-style-type: none"> <li>• Para la Practica 3: Imprimir en pantalla la alternativa correcta dependiendo de una acción cuando se cumplan dos determinadas condiciones.</li> <li>• Para la Practica 4: Imprimir en pantalla la solución correcta a la función: <math>y = x^n</math> (potencia n de x).</li> <li>• Para ambas prácticas, resolver problemas reales y cotidianos del entorno.</li> </ul> <p>Criterios de Fondo (parte técnica):</p> <ul style="list-style-type: none"> <li>• Para ambas prácticas, ejecutarse correctamente sin errores en tiempo de ejecución, ni de compilación.</li> <li>• Para ambas prácticas, el código estará indentado y comentado.</li> </ul>	30 %
	<p>Desarrolla programas utilizando arreglos o arrays mediante la manipulación de estructuras básicas de datos para ordenación y búsqueda, de manera analítica, eficiente y creativa.</p>	<p>Reporte de la Practica 5 Programa utilizando “Arreglos”</p> <p>Reporte de la Practica 6 Programa utilizando “Ordenación”</p>	<p>Instrumento de Evaluación:</p> <ul style="list-style-type: none"> <li>• Lista de cotejo</li> </ul> <p>Criterios de Forma (estilo):</p> <ul style="list-style-type: none"> <li>• Para la Practica 5: Imprimir en pantalla un arreglo multidimensional de forma correcta.</li> <li>• Para la Practica 6: Imprimir en pantalla la ordenación de un arreglo mediante el método de la burbuja de forma correcta.</li> <li>• Para ambas prácticas, resolver problemas reales y cotidianos del entorno.</li> </ul>	



**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

3			<p>Criterios de Fondo (parte técnica):</p> <ul style="list-style-type: none"> <li>• Para ambas prácticas, ejecutarse correctamente sin errores en tiempo de ejecución, ni de compilación.</li> <li>• Para ambas prácticas, el código estará indentado y comentado.</li> </ul>	40 %
Propósito de la Unidad de Aprendizaje		Evidencia Integradora	Criterios e Instrumento de Evaluación	Porcentaje de Acreditación
<p>Construye programas informáticos en un lenguaje de programación, para resolver problemas de informática, con base en los algoritmos, en las estructuras de selección, repetición, en los arreglos unidimensionales, multidimensionales, en la ordenación y búsqueda en listas de forma analítica, creativa e innovadora, considerando un enfoque de desarrollo práctico y sustentable.</p>		<p>Codificación de 20 programas como práctica integradora en los que se utilicen estructuras selectivas, sobrecarga de funciones, arreglos multidimensionales y ordenación por selección para la solución de problemas reales en su entorno de una forma individual.</p>	<p>Instrumento de Evaluación:</p> <ul style="list-style-type: none"> <li>• Lista de cotejo</li> </ul> <p>Criterios de Forma (estilo):</p> <ul style="list-style-type: none"> <li>• Todos los programas se deben entregar de forma individual en una carpeta digital divididos por temas según cada unidad temática.</li> <li>• Todos los programas deben imprimir en pantalla la/las soluciones(es) de forma individual.</li> <li>• Todos los programas deben resolver problemas reales y cotidianos del entorno.</li> </ul> <p>Criterios de Fondo (parte técnica):</p> <ul style="list-style-type: none"> <li>• Todos los programas deben, ejecutarse correctamente sin errores en tiempo de ejecución, ni de compilación.</li> <li>• Todos los programas deben tener el código indentado y comentado.</li> </ul>	100%





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

**PROGRAMA SINTÉTICO**

**PROPÓSITO DE LA UNIDAD DE APRENDIZAJE**

Construye programas informáticos en un lenguaje de programación, para resolver problemas de informática, con base en los algoritmos, en las estructuras de selección, repetición, en los arreglos unidimensionales, multidimensionales, en la ordenación y búsqueda en listas de forma analítica, creativa e innovadora, considerando un enfoque de desarrollo práctico y sustentable.

N°	UNIDAD DE COMPETENCIA	APRENDIZAJES ESPERADOS	CONTENIDOS DE APRENDIZAJE/SABERES
1	Relaciona los conceptos y paradigmas de programación, utilizando herramientas para el aprendizaje de los procesos de compilación y ejecución de los programas con base en un entorno de desarrollo integrado (IDE) de forma innovadora, eficiente y creativa.	<p>1.1. Domina el concepto de la programación, mediante la utilización de sus características y elementos principales, con base al pensamiento lógico y a la solución a diversos problemas y paradigmas de la programación de una forma individual y creativa.</p> <p>1.2. Ejecuta programas mediante un editor y compilador independientes, en un entorno de desarrollo integrado (IDE) para la resolución de problemas diversos con un enfoque de innovación, eficiente y creativo.</p>	<p><b>CONCEPTUALES</b> Conceptos básicos de programación Datos, tipos de datos y operaciones primitivas Paradigmas de programación Entorno de desarrollo IDE Operación de asignación Estructura de programas</p> <p><b>PROCEDIMENTALES</b> Utiliza las herramientas de programación básicas incluidas en el sistema operativo para la ejecución de programas. Utiliza de manera eficiente el entorno de desarrollo integrado (IDE) para la elaboración de los programas informáticos.</p> <p><b>ACTITUDINALES</b> Atiende una actitud respetuosa hacia sus compañeros y docentes, el desarrollo de sus actividades las realiza de una forma analítica, reflexiva, con una comunicación efectiva y el trabajo se cumple de forma colaborativa.</p>
2			<p><b>CONCEPTUALES</b> Expresiones Reglas de prioridad Funciones internas Conversión de tipo de datos El flujo de control de un programa Estructuras secuenciales Estructuras selectivas La sentencia ir-a (goto)</p>





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

	<p>Desarrolla programas considerando los elementos básicos que lo componen, mediante operaciones de entrada - salida para resolver problemas comunes de informática con un enfoque creativo, de innovación y de trabajo en equipo</p>	<p>2.1. Codifica programas utilizando estructuras de control condicionales y selectivas para aumentar su funcionalidad, mediante los tipos primitivos de datos, los operadores aritméticos y la conversión de tipos de datos, construyendo funciones de manera creativa, asertiva y con innovación.</p> <p>2.2. Construye programas que implementen funciones para resolver problemas comunes de informática con un enfoque creativo, de innovación y de trabajo en equipo.</p>	<p>Estructuras Repetitivas  Estructuras Repetitivas anidadas  Funciones  Procedimientos (subrutinas)  Ámbito: variables locales y globales  Comunicación con subprogramas  Recursión (recursividad)  Sobrecarga de funciones</p> <p><b>PROCEDIMENTALES</b>  Construye programas utilizando operadores, variables, constantes, tipos primitivos de datos, operadores aritméticos y conversión de tipos de datos, para resolver problemas comunes de informática.  Ejecuta instrucciones que incluyen estructuras de condición, de selección, de repetición para resolver problemas comunes de informática.  Crea funciones para resolver problemas comunes de informática  Utiliza procedimientos y subrutinas para la comunicación entre programas y resolver problemas comunes de informática.  Realiza programas que utilizan recursividad para resolver problemas comunes de informática.</p> <p><b>ACTITUDINALES</b>  Se interesa en la resolución de problemas comunes de informática de una manera creativa, asertiva e innovadora.  Respeto el trabajo de sus compañeros y docentes permitiendo el desarrollo de las actividades de una manera colaborativa.</p>
3			<p><b>CONCEPTUALES</b>  Estructuras de datos  Arrays (arreglos) unidimensionales: los vectores  Operaciones con vectores  Arrays de varias dimensiones  Arrays multidimensionales  Almacenamiento de arrays en memoria  Estructuras versus registros  Arrays de estructuras  Uniones  Unión versus estructura  Enumeraciones</p>







**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

	<p>Desarrolla programas utilizando arreglos o arrays mediante la manipulación de estructuras básicas de datos para ordenación y búsqueda, de manera analítica, eficiente y creativa.</p>	<p>3.1. Construye programas que utilizan arreglos unidimensionales y multidimensionales mediante las estructuras de datos, operación con vectores para solucionar y resolver problemas reales y cotidianos con innovación, adaptabilidad, empatía y proactividad.</p> <p>3.2. Construye programas que utilizan uniones, enumeraciones, ordenación de listas y búsquedas secuenciales para solucionar y resolver problemas reales y cotidianos, con innovación, adaptabilidad, empatía y proactividad.</p>	<p>Ordenación          Búsqueda          Intercalación</p> <p><b>PROCEDIMENTALES</b></p> <p>Utiliza las estructuras de datos, operación con vectores para realizar ordenación y búsqueda de elementos.          Aplica arrays unidimensionales y multidimensionales para resolver problemas reales y cotidianos.          Utiliza uniones versus estructuras para la resolución de problemas informáticos comunes.          Realiza programas que utilizan la ordenación y las búsquedas para resolver problemas reales y cotidianos de su entorno.</p> <p><b>ACTITUDINALES</b></p> <p>Respeto las ideas y opiniones de sus compañeros y docentes.          Se interesa por la resolución de los problemas encontrados en su entorno con innovación, adaptabilidad, empatía y proactividad.          Atiende las problemáticas de su entorno, trabajando con una comunicación asertiva y de forma empática.</p>
--	--	---	---





**Programa Académico: Técnico en Programación**

**Unidad de Aprendizaje: Programación Básica**

**BIBLIOGRAFÍA BÁSICA Y COMPLEMENTARIA**

Número y Nombre de la Unidad Didáctica	FORMATO APA	CLASIFICACIÓN	
		Básico	Consulta
Unidad didáctica 1: Introducción a la programación	Aguilar, Joyanes. Luis (2022, 24 octubre). Fundamentos De Programación (4.a ed.). MCGRAW HILL EDDUCATION.	X	
Unidad didáctica 1: Introducción a la programación	Buriticá, O. I. T. & U, E. de la. (2022, 24 octubre). Lógica de programación (1.a ed.). Ediciones de la U		X
Unidad didáctica 2: Elementos básicos de la programación	Aguilar, Joyanes Luis. (2022, 24 octubre). Fundamentos De Programación (4.a ed.). MCGRAW HILL EDDUCATION.	X	
Unidad didáctica 2: Elementos básicos de la programación	Sierra, Francisco Javier Ceballos. (2022, 24 octubre). Java 2. Curso de Programación. 4a Edición. Alianza Editorial.		X
Unidad didáctica 3: Arreglos y estructuras de datos	Aguilar, Joyanes Luis. (2022, 24 octubre). Fundamentos De Programación (4.a ed.). MCGRAW HILL EDDUCATION	X	
Unidad didáctica 3: Arreglos y estructuras de datos	Zahonero, Martínez Ignacio. (2022, 24 octubre). Estructura de datos en Java (4.a ed.). MCGRAW HILL EDDUCATION.		X

