

 <b>UNIVERSIDAD</b> <b>SERGIO ARBOLEDA</b>	<b>GESTIÓN ACADÉMICA</b>	<b>CÓDIGO:</b> <b>ACA-FO-96</b>
		<b>VERSIÓN:</b> 03
	<b>FORMATO CONTENIDO PROGRAMÁTICO DE LA ASIGNATURA</b> <b>PROGRAMAS DE PREGRADO Y POSTGRADO</b> <b>(FORMACIÓN POR COMPETENCIAS)</b>	<b>FECHA:</b> 27/09/2024

Escuela		Nombre del Programa o Departamento		Modalidad del programa			Versión del plan de estudios
Escuela de Ciencias Exactas e Ingeniería		Ciencias de la Computación e Inteligencia Artificial		Presencial			Versión 1
Nombre de la asignatura/Módulo Estructuras de Datos Lineales		Modalidad de la asignatura (seleccione) Presencial		Semestre (pregrados)	I	Cohorte (postgrados)	
Área o componente de formación	Básico	Tipo de asignatura		Obligatoria	Código	SIST0027	Créditos
Horas presenciales	4	Horas sincrónicas		Horas asincrónicas		Horas de trabajo autónomo	5

Versión CPA	Fecha de actualización del CPA
1	19/12/2024

#### JUSTIFICACIÓN Y DESCRIPCIÓN DE LA ASIGNATURA

El estudio de las Estructuras de Datos Lineales es fundamental en la formación de un estudiante de la computación e inteligencia artificial por las siguientes razones:

- Base para la organización y manipulación eficiente de la información:** Las estructuras de datos lineales, como arreglos, listas enlazadas, pilas y colas, proporcionan los cimientos para comprender cómo se organiza y manipula la información dentro de un programa. Esta comprensión es crucial para desarrollar algoritmos eficientes y optimizar el rendimiento de las aplicaciones.
- Resolución de problemas computacionales:** Muchos problemas en ciencias de la computación e inteligencia artificial se resuelven mediante el uso adecuado de estructuras de datos lineales. La elección de la estructura correcta puede simplificar la solución y mejorar su eficiencia.
- Fundamento para estructuras de datos más complejas:** Las estructuras de datos lineales sirven como base para comprender estructuras de datos más complejas, como árboles, grafos y tablas hash, que son esenciales en áreas como bases de datos, redes y aprendizaje automático.
- Desarrollo del pensamiento algorítmico:** El estudio de las estructuras de datos lineales fomenta el desarrollo del pensamiento algorítmico y la capacidad de abstracción, habilidades cruciales para la resolución de problemas en cualquier área de la computación.

En el contexto de las Ciencias de la Computación e Inteligencia Artificial, las estructuras de datos lineales son relevantes en diversas áreas:

 <b>UNIVERSIDAD SERGIO ARBOLEDA</b>	<b>GESTIÓN ACADÉMICA</b>	<b>CÓDIGO:</b> <b>ACA-FO-96</b>
		<b>VERSIÓN: 03</b>
	<b>FORMATO CONTENIDO PROGRAMÁTICO DE LA ASIGNATURA PROGRAMAS DE PREGRADO Y POSTGRADO (FORMACIÓN POR COMPETENCIAS)</b>	<b>FECHA: 27/09/2024</b>

## JUSTIFICACIÓN Y DESCRIPCIÓN DE LA ASIGNATURA

- **Implementación de algoritmos:** Muchos algoritmos fundamentales, como los de búsqueda y ordenamiento, se implementan utilizando estructuras de datos lineales.
- **Representación de datos:** Las estructuras de datos lineales se utilizan para representar datos en diversas aplicaciones, como el procesamiento de texto, el análisis de secuencias genéticas y la gestión de datos en aplicaciones web.

## UNIDADES DE COMPETENCIA

El Programa de Ciencias de la Computación e Inteligencia Artificial de la Universidad Sergio Arboleda busca desarrollar en sus estudiantes las competencias propuestas por el organismo internacional ABET Inc, el cual otorga acreditación a los programas de educación superior en Ciencias Aplicadas, Informática, Ingenierías y Tecnología. A continuación, se indican las competencias que serán trabajadas durante el curso. En la siguiente lista se explican las competencias y unidades de competencia que se desarrollan en el curso:

### Competencias

1. Identificar, formular y resolver problemas complejos de ingeniería mediante la aplicación de principios de ingeniería, ciencia y matemáticas.
2. Habilidad de aplicar el diseño de ingeniería para producir soluciones que satisfagan necesidades específicas con consideración de la salud pública, seguridad y el bienestar, así como factores globales, culturales, sociales, ambientales y económicos.

### Unidades de competencia

- 1.3 Resolver problemas y Optimizar soluciones de Ingeniería en Contextos Específicos.
- 2.1 Diseñar Soluciones de Ingeniería con enfoques culturales, sociales, económicas y ambientales.
- 2.2 Desarrollar Soluciones de Ingeniería con Enfoque en Factores Globales y Sostenibilidad.

### Alcance formativo

- 1.3 La unidad pretende que el estudiante comprenda que son las estructuras de datos dinámicas lineales, cuáles son las operaciones básicas y su aplicación en la solución a problemas de datos.
- 2.1 La unidad permite que el estudiante entienda qué es una estructura de datos dinámica lineales colas, cómo funciona y cuáles son sus principales operaciones. Distinguir los diferentes tipos de colas. Además, resolver problemas algorítmicos clásicos utilizando estructuras de datos lineales colas.
- 2.2 Esta unidad tiene como propósito introducir al estudiante en los conceptos y fundamentos de ordenamiento y búsqueda de datos, así como en la implementación de algoritmos de ordenamiento y búsqueda para estructuras de datos lineales estáticas y dinámicas.

- **Inteligencia artificial:** En áreas como el procesamiento del lenguaje natural, las listas y colas son útiles para gestionar secuencias de palabras o eventos. En la planificación y búsqueda, las colas se utilizan en algoritmos de búsqueda en anchura.
- **Desarrollo de software:** El conocimiento de las estructuras de datos lineales es esencial para el desarrollo de software eficiente y robusto.

 <b>UNIVERSIDAD SERGIO ARBOLEDA</b>	<b>GESTIÓN ACADÉMICA</b>	<b>CÓDIGO:</b> <b>ACA-FO-96</b>
		<b>VERSIÓN: 03</b>
	<b>FORMATO CONTENIDO PROGRAMÁTICO DE LA ASIGNATURA PROGRAMAS DE PREGRADO Y POSTGRADO (FORMACIÓN POR COMPETENCIAS)</b>	<b>FECHA: 27/09/2024</b>

## JUSTIFICACIÓN Y DESCRIPCIÓN DE LA ASIGNATURA

El estudio de las Estructuras de Datos Lineales es crucial para la formación de un profesional en Ciencias de la Computación e Inteligencia Artificial, proporcionando las bases para la organización y manipulación eficiente de la información, la resolución de problemas computacionales y el desarrollo de algoritmos eficientes. La asignatura aborda los conceptos teóricos, la implementación práctica, el análisis de eficiencia y las aplicaciones en el contexto de la computación e inteligencia artificial.

## RESULTADOS DE APRENDIZAJE

Los resultados de aprendizaje son enunciados que describen los conocimientos, habilidades y actitudes específicos que los estudiantes deben demostrar al finalizar la asignatura. Estos resultados deben ser medibles, observables y evaluables, y deben estar directamente relacionados con las unidades de competencia aterrizzadas en el contexto del curso.

### Resultados de aprendizaje de la U.C. 1.3:

- Entender qué es una estructura de datos dinámica Lista, cómo funciona y cuáles son sus principales operaciones.
- Diseñar, implementar y analizar algoritmos que utilicen estructura de datos dinámica lista.
- Entender que son las listas encadenadas, listas ordenadas, doblemente encadenadas y lista circulares.
- Entender qué es una estructura de datos dinámica Pila, cómo funciona y cuáles son sus principales operaciones.
- Diseñar, implementar y analizar algoritmos que utilicen estructura de datos dinámicas pilas.
- Reconocer los diferentes escenarios donde las estructuras de datos dinámicas pilas son útiles.
- Aplicar las operaciones básicas de las estructuras de datos dinámicas Pilas para dar solución a problemas con la técnica de programación recursión.
- Identifica patrones repetitivos en la solución de problemas.

### Resultados de aprendizaje de la U.C. 2.1:

- Entender qué es una cola, cómo funciona y cuáles son sus principales operaciones (Enqueue, Dequeue, Front).
- Modelar problemas del mundo real utilizando el concepto de estructura de datos dinámicas cola.
- Diseñar, implementar y analizar algoritmos que utilicen estructura de datos dinámicas cola.
- Reconocer los diferentes escenarios donde las estructuras de datos dinámicas colas son útiles.
- Entender que son las colas de prioridad, colas circulares y bicolas.

### Resultados de aprendizaje de la U.C. 2.2:

- Conocer y aplicar los principales algoritmos de ordenamiento (burbuja, inserción, selección, quicksort, mergesort, Heap sort, Counting sort, Radix sort, Bucket sort) y búsqueda (secuencial, binaria).
- Evaluar la eficiencia de diferentes algoritmos en términos de tiempo y espacio.

 UNIVERSIDAD SERGIO ARBOLEDA	<b>GESTIÓN ACADÉMICA</b> <b>FORMATO CONTENIDO PROGRAMÁTICO DE LA ASIGNATURA</b> <b>PROGRAMAS DE PREGRADO Y POSTGRADO</b> <b>(FORMACIÓN POR COMPETENCIAS)</b>	CÓDIGO: ACA-FO-96  VERSIÓN: 03  FECHA: 27/09/2024
---	---	--

## RESULTADOS DE APRENDIZAJE

- Reconocer los diferentes escenarios donde los algoritmos de ordenamiento y búsqueda son útiles.
- Reconocer y aplicar las estructuras de datos Tablas Hash.
- Evaluar la eficiencia de diferentes algoritmos en términos de tiempo y espacio.
- Reconocer los diferentes escenarios donde los algoritmos de ordenamiento y búsqueda son útiles.
- Reconocer y aplicar las estructuras de datos Tablas Hash.

## CONTENIDOS Y/O ACTIVIDADES

### • Semana 1

**Tema:** Presentación y reglas del curso, entrega de contenido programático y firma de acta de compromiso.

**Tema:** Tipos de datos abstractos (TDA): Definición y la importancia de la abstracción en la programación.

Concepto de estructura de datos: Organización y manipulación de datos en la memoria. Clasificación de las estructuras de datos: Lineales vs. no lineales.

### • Semana 2

**Tema:** Estructuras de Datos Lineales Fundamentales: Arreglos (Arrays), Arreglos unidimensionales y multidimensionales. Operaciones básicas: acceso, inserción, eliminación, búsqueda.

**Tema:** Estructuras de Datos Lineales Fundamentales: Listas Enlazadas (Linked Lists): Listas simplemente enlazadas. Operaciones básicas: inserción, eliminación (al principio, al final, en una posición específica), recorrido.

### • Semana 3

**Tema:** Estructuras de Datos Lineales Fundamentales: Listas Enlazadas (Linked Lists): Listas doblemente enlazadas. Operaciones básicas: inserción, eliminación (al principio, al final, en una posición específica), recorrido.

**Tema:** Estructuras de Datos Lineales Fundamentales: Listas Enlazadas (Linked Lists): Listas circulares Operaciones básicas: inserción, eliminación (al principio, al final, en una posición específica), recorrido.

### • Semana 4

**Tema:** Estructuras de Datos Lineales Fundamentales: Pilas (Stacks): Principio LIFO (Last-In, First-Out). Operaciones push (apilar) y pop (desapilar). Aplicaciones.

**Tema:** Estructuras de Datos Lineales Fundamentales: Pilas (Stacks): Implementaciones con arreglos y listas enlazadas.

### • Semana 5

**Tema:** Recursión, Definición y Concepto Básico, Definición recursiva, Componentes Clave de una Función Recursiva.

 <b>UNIVERSIDAD SERGIO ARBOLEDA</b>	<b>GESTIÓN ACADÉMICA</b>	<b>CÓDIGO:</b> <b>ACA-FO-96</b>
		<b>VERSIÓN: 03</b>
	<b>FORMATO CONTENIDO PROGRAMÁTICO DE LA ASIGNATURA PROGRAMAS DE PREGRADO Y POSTGRADO (FORMACIÓN POR COMPETENCIAS)</b>	<b>FECHA: 27/09/2024</b>

**Tema:** Funcionamiento de la Recursión, pila de llamadas (Call Stack). Tipos de Recursión. Aplicaciones

- **Semana 6**

**Tema:** Funcionamiento de la Recursión, pila de llamadas (Call Stack), aplicaciones.

**Tema:** Estructuras de Datos Lineales Fundamentales: Colas (Queues): Principio FIFO (First-In, First-Out). Operaciones enqueue (encolar) y dequeue (desencolar).

- **Semana 7**

**Tema:** Estructuras de Datos Lineales Fundamentales: Colas (Queues). Aplicaciones.

**Tema:** Estructuras de Datos Lineales Fundamentales: Colas circulares. Aplicaciones.

- **Semana 8**

**Tema:** Estructuras de Datos Lineales Fundamentales: Colas de prioridad. Aplicaciones.

**Tema:** Estructuras de Datos Lineales Fundamentales: Doble cola (deque - double-ended queue). Aplicaciones.

- **Semana 9**

**Tema:** Implementaciones de Colas: Con arreglos (arrays) y Con Lista enlazadas.

**Tema:** Complejidad algorítmica, Notación Asintótica, Notaciones principales: Big O ( $O$ ), Omega ( $\Omega$ ), Theta ( $\Theta$ ).

- **Semana 10**

**Tema:** Reglas para calcular la complejidad Big O, Análisis de la Complejidad mejor, peor y caso promedio.

**Tema:** Análisis de la Complejidad mejor, peor y caso promedio, Análisis de la eficiencia de las operaciones en las estructuras de datos (tiempo y espacio).

- **Semana 11**

**Tema:** Algoritmos de ordenamiento. Definición y Concepto Básico. Tipos de Algoritmos de Ordenamiento.

**Tema:** Algoritmos de ordenamiento básicos (simples): Insertion Sort, Selection Sort, Bubble Sort.

- **Semana 12**

**Tema:** Algoritmos de ordenamiento eficientes (avanzados): Merge Sort, Quick Sort, Heap Sort.

**Tema:** Otros algoritmos de ordenamiento: Radix Sort (Ordenamiento de raíz), Bucket Sort (Ordenamiento por casilleros)

- **Semana 13**

**Tema:** Análisis de la Complejidad Algorítmica de los algoritmos de ordenamiento.

**Tema:** Algoritmos de búsqueda. Definición y Concepto Básico, Tipos de Algoritmos de Búsqueda.

- **Semana 14**

**Tema:** Tipos de Algoritmos de Búsqueda: Búsqueda lineal (o secuencial), Búsqueda binaria.

 <b>UNIVERSIDAD SERGIO ARBOLEDA</b>	<b>GESTIÓN ACADÉMICA</b>	<b>CÓDIGO:</b> <b>ACA-FO-96</b>
	<b>FORMATO CONTENIDO PROGRAMÁTICO DE LA ASIGNATURA PROGRAMAS DE PREGRADO Y POSTGRADO (FORMACIÓN POR COMPETENCIAS)</b>	<b>VERSIÓN: 03</b>
		<b>FECHA: 27/09/2024</b>

**Tema:** Tipos de Algoritmos de Búsqueda: Búsqueda en tablas hash.

- **Semana 15**

**Tema:** Análisis de la Complejidad Algorítmica de búsqueda.

**Tema:** Tipos de Algoritmos de Búsqueda: Búsqueda por interpolación, Búsqueda en árboles.

- **Semana 16**

**Tema:** Trabajo de aula en equipos de trabajo conformado por estudiantes, donde se diseñará e implementará un módulo de software que debe realizar una simulación en un escenario definido. El software de implementar estructuras de datos dinámicas pilas, colas, listas y algoritmos de ordenamiento y búsqueda, además de técnicas de programación como la recursión.

**Tema:** Trabajo de aula en equipos de trabajo conformado por estudiantes, donde se diseñará e implementará un módulo de software que debe realizar una simulación en un escenario definido. El software de implementar estructuras de datos dinámicas pilas, colas, listas y algoritmos de ordenamiento y búsqueda, además de técnicas de programación como la recursión.

### **SISTEMA DE EVALUACIÓN (pregrados)**

Resultado de aprendizaje	Criterios de evaluación	Evidencia de aprendizaje	Evaluador (modalidades de evaluación)	Instrumentos
<b>R.A. U.C. 1.3</b>	<p>El código implementado para dar solución al problema debe ser claro, conciso, bien estructurado y la declaración de las estructuras de datos dinámicas pilas muy bien definidas.</p> <p>. En el código implementado la declaración de las</p>	<p>Preguntas teóricas y prácticas para evaluar los principios y la aplicación de las estructuras de datos dinámicas listas.</p> <p>Talleres donde el estudiante aplica las operaciones básicas y comprender su impacto en la</p>	<p>Heteroevaluación Autoevaluación</p>	Rúbrica del docente

 UNIVERSIDAD SERGIO ARBOLEDA	<b>GESTIÓN ACADÉMICA</b> <b>FORMATO CONTENIDO PROGRAMÁTICO DE LA ASIGNATURA</b> <b>PROGRAMAS DE PREGRADO Y POSTGRADO</b> <b>(FORMACIÓN POR COMPETENCIAS)</b>	CÓDIGO: ACA-FO-96  VERSIÓN: 03  FECHA: 27/09/2024
---	---	--

	<p>estructuras recursivas al igual que las variables deben estar bien estructuradas para producir los resultados esperados para todos los casos de prueba.</p> <ul style="list-style-type: none"> <li>. El código debe estar bien documentado para facilitar su comprensión.</li> <li>. El estudiante debe ser capaz de explicar el funcionamiento de su código y analizar su complejidad.</li> </ul>	<p>estructura de las listas.</p> <p>Talleres donde el estudiante aplica las operaciones básicas de una pila utilizando una estructura de datos estática (arrays).</p> <p>Preguntas teóricas y prácticas para evaluar los principios y la aplicación de las estructuras de datos dinámicas pilas.</p> <p>Talleres donde el estudiante aplica las operaciones básicas y comprender su impacto en la estructura de la pila.</p> <p>Talleres donde el estudiante aplica las operaciones básicas de una pila utilizando una estructura de datos estática (arrays).</p>		
R.A. U.C. 2.1	El código	Preguntas teóricas y	Heteroevaluación	Rúbrica del docente

 UNIVERSIDAD SERGIO ARBOLEDA	<b>GESTIÓN ACADÉMICA</b> <b>FORMATO CONTENIDO PROGRAMÁTICO DE LA ASIGNATURA</b> <b>PROGRAMAS DE PREGRADO Y POSTGRADO</b> <b>(FORMACIÓN POR COMPETENCIAS)</b>	CÓDIGO: ACA-FO-96  VERSIÓN: 03  FECHA: 27/09/2024
---	---	--

	<p>implementado para dar solución al problema debe ser claro, conciso, bien estructurado y la declaración de las estructuras de datos dinámicas Colas muy bien definidas.</p> <p>En el código implementado la declaración de las estructuras recursivas al igual que las variables deben estar bien estructura para producir los resultados esperados para todos los casos de prueba.</p> <p>. El código debe estar bien documentado para facilitar su comprensión.</p>	<p>prácticas para evaluar la comprensión de conceptos y la capacidad de resolver problemas implementando estructuras de datos dinámicas colas.</p> <p>Talleres de diseño y programación donde los estudiantes implementen algoritmos utilizando colas y recursión.</p> <p>Análisis de los resultados obtenidos, comparando diferentes algoritmos.</p> <p>Talleres de diseño y programación donde los estudiantes implementen algoritmos utilizando colas, pilas, listas y recursión.</p> <p>Presentaciones orales donde los</p>	Autoevaluación	
--	---	---	----------------	--

 UNIVERSIDAD SERGIO ARBOLEDA	<b>GESTIÓN ACADÉMICA</b> <b>FORMATO CONTENIDO PROGRAMÁTICO DE LA ASIGNATURA</b> <b>PROGRAMAS DE PREGRADO Y POSTGRADO</b> <b>(FORMACIÓN POR COMPETENCIAS)</b>	CÓDIGO: ACA-FO-96  VERSIÓN: 03  FECHA: 27/09/2024
---	---	--

		<p>estudiantes explican sus soluciones a problemas y justifican sus decisiones de diseño.</p> <p>Análisis de los resultados obtenidos, comparando diferentes algoritmos.</p>		
<b>R.A. U.C. 2.2</b>	<p>El código implementado para dar solución al problema debe ser claro, conciso, bien estructurado y debe ser eficientes en términos de tiempo y espacio.</p> <p>En el código implementado la selección de las estructuras de datos dinámicas pilas, colas y listas, así como los algoritmos de ordenamiento y búsqueda debe estar bien estructurados para producir los resultados esperados para todos los casos de</p>	<p>Elaborar un trabajo de aula guiado en equipos de trabajo conformado por estudiantes, donde se diseñara e implementara un modulo de software que debe realizar una simulación en un escenario definido (cajero electrónico, cajas supermercado, sistema operativo, rutas y transportadores, etc).</p> <p>El modulo de software debe implementar estructuras de datos</p>	Heteroevaluación Autoevaluación	Rúbrica del docente

 UNIVERSIDAD SERGIO ARBOLEDA	<b>GESTIÓN ACADÉMICA</b>	CÓDIGO: ACA-FO-96
		VERSIÓN: 03
	<b>FORMATO CONTENIDO PROGRAMÁTICO DE LA ASIGNATURA            PROGRAMAS DE PREGRADO Y POSTGRADO            (FORMACIÓN POR COMPETENCIAS)</b>	FECHA: 27/09/2024

	<p>prueba.</p> <p>El código debe estar bien documentado para facilitar su comprensión.</p> <p>El estudiante debe ser capaz de explicar el funcionamiento de su código y analizar su complejidad.</p>	<p>dinámicas pilas, colas, listas y algoritmos de ordenamiento y búsqueda, además de técnicas de programación como la recursión.</p>		
--	--	--	--	--

#### SISTEMA DE EVALUACIÓN

Corte	Estrategia sumativa (evidencia de aprendizaje)	Unidades de competencia (UC)			% Corte
		1.3	2.1	2.2	
	Talleres donde el estudiante aplica las operaciones básicas y comprender su impacto en la estructura de las listas	10%			

 UNIVERSIDAD SERGIO ARBOLEDA	<b>GESTIÓN ACADÉMICA</b>	CÓDIGO: ACA-FO-96
	<b>FORMATO CONTENIDO PROGRAMÁTICO DE LA ASIGNATURA PROGRAMAS DE PREGRADO Y POSTGRADO (FORMACIÓN POR COMPETENCIAS)</b>	VERSIÓN: 03  FECHA: 27/09/2024



 <b>UNIVERSIDAD SERGIO ARBOLEDA</b>	<b>GESTIÓN ACADÉMICA</b>	<b>CÓDIGO:</b> <b>ACA-FO-96</b>
	<b>FORMATO CONTENIDO PROGRAMÁTICO DE LA ASIGNATURA PROGRAMAS DE PREGRADO Y POSTGRADO (FORMACIÓN POR COMPETENCIAS)</b>	<b>VERSIÓN: 03</b>
		<b>FECHA: 27/09/2024</b>

<b>PERFIL DEL DOCENTE</b>		
Profesional en	Ciencias de la computación, Ingeniería de Sistemas, Ingeniería Informática o afines.	Otra habilidad o competencia específica requerida:  Análisis y diseño orientado a objetos y experiencia y lenguaje de programación orientado a objetos, experiencia en estructuras de datos lineales y no lineales.
Postgrado en	Maestría.	
Años experiencia profesional (preferiblemente)	3	

<b>BIBLIOGRAFÍA</b>	
ESTRUCTURAS DE DATOS Y ALGORITMOS, Roberto Hernandez, Juan Carlos Lázaro, PEARSON EDUCACION, 2000	
ESTRUCTURAS DE DATOS, Osvaldo Cairó, Silvia Giardati, McGraw Hill.	
ESTRUCTURA DE DATOS, Luis Joyanes Aguilar, McGraw-Hill	