

Asignatura: Algoritmos y Estructuras de Datos

Guía Didáctica No. 4

Semana del 7 al 11 de abril de 2025

Colectivo de Asignatura

Managua, 6 de abril 2025

Algoritmos y Estructuras de Datos

I. CONTENIDOS

Unidad II: Estructuras de datos lineales

- 1. Listas enlazadas
 - 1.1 Introducción a las listas enlazadas
 - 1.2 Operaciones sobre listas

II. LOGROS DE APRENDIZAJES

Al finalizar la semana, los estudiantes serán capaces de:

- 1. Comprenderán el concepto de listas enlazadas, incluyendo su estructura y ventajas sobre los arrays.
- 2. Podrán describir los componentes de una lista enlazada e implementar una lista enlazada simple en código.
- 3. Aprenderán a realizar operaciones de inserción y eliminación en listas enlazadas.
- 4. Explorarán aplicaciones del mundo real de las listas enlazadas y las compararán con otras estructuras de datos.

III. MATERIAL DE ESTUDIO Y RECURSOS

- Material de lectura
 - https://docs.python.org/es/3.13/tutorial/index.html
 - Cuevas, A. (2016). Capítulo 4: Programación Funcional. En Python 3: Curso práctico (1^{ra.} Ed., pp. 145 179). RA-MA Editorial
 - Cuevas, A. (2016). Capítulo 5: Programación Orientada a Objetos. En Python 3: Curso práctico (1^{ra.} Ed., pp. 183 – 240). RA-MA Editorial
- Recursos
 - Visual Studio Code Python
 - Git GitHub
 - UAM Virtual

ORIENTACIONES METODOLÓGICAS

Para alcanzar los logros de aprendizajes planteados, trabajaremos con las siguientes estrategias y recursos:



Actividad de Aprendizaje No. 1 – Examen Teórico-Práctico - Corte 1

Tiempo de inversión : 2 horas Periodo de ejecución : 7 de abril Modalidad : individual

Puntaje : actividad sumativa

Descripción de la actividad:

Examen de la Unidad I: Introducción al Lenguaje de Programación Python.

Actividad de Aprendizaje No. 2 - Introducción a las Listas Enlazadas

Tiempo de inversión :1 hora Periodo de ejecución : 7 de abril Modalidad : grupal

: actividad formativa Puntaie

Descripción de la actividad:

Definición de listas enlazadas, nodos y enlaces.

Diferencias entre listas enlazadas y arrays.

Demostración visual de la estructura de una lista enlazada utilizando diagramas. Discusión grupal sobre aplicaciones prácticas de las listas enlazadas en la ingeniería de software.

Actividad de Aprendizaje No. 3 – Estructura de una lista enlazada

Tiempo de inversión :1 hora Periodo de ejecución : 9 de abril Modalidad : grupal

: actividad formativa Puntaje

Descripción de la actividad:

Componentes de una lista enlazada: cabeza (head), nodos, cola (tail). Operaciones básicas de listas enlazadas: inserción y eliminación. Evaluación: Entrega del código de la lista enlazada con comentarios explicando cada parte.

Recursos de trabajo:

- Plataforma virtual
- Visual Studio Code Python
- Git GitHub

Actividad de Aprendizaje No. 4 – Operaciones sobre Listas Enlazadas

Tiempo de inversión : 1 hora (casa)
Periodo de ejecución : 9 de abril
Modalidad : individual

Puntaje : actividad formativa

Descripción de la actividad:

Los estudiantes realizan una lectura del material proporcionado por el docente, Conferencia sobre algoritmos de inserción y eliminación, diferentes tipos de inserción (al principio, en el medio, al final), replicando los ejercicios resueltos.

Métodos y consideraciones para la eliminación.

- Cuevas, A.(2016). Capítulo 4: Programación Funcional. En Python 3: Curso práctico (1^{ra.} Ed., pp. 145 179). RA-MA Editorial
- Cuevas, A. (2016). Capítulo 5: Programación Orientada a Objetos. En Python 3: Curso práctico (1^{ra.} Ed., pp. 183 – 240). RA-MA Editorial

Actividad de Aprendizaje No. 5 – Aplicaciones prácticas de Listas Enlazadas

Tiempo de inversión : 4 horas (casa) Periodo de ejecución : 7 al 11 abril

Modalidad: trabajo en grupoPuntaje: actividad sumativa

Descripción de la actividad:

Análisis de casos de estudio de aplicaciones que utilizan listas enlazadas (por ejemplo, gestión de memoria).

Lluvia de ideas en grupo sobre dónde se pueden aplicar las listas enlazadas en el desarrollo de software.

Evaluación:

Presentación grupal sobre una aplicación seleccionada de listas enlazadas, que contemple:

- Casos de uso de listas enlazadas en varios escenarios de programación.
- Análisis de rendimiento en comparación con otras estructuras de datos.

Recursos de trabajo:

- Plataforma virtual
- Visual Studio Code Python
- Git GitHub

Guía de Ejercicios Prácticos

- Implementar un método que recibe una lista de enteros L y un número entero n de forma que modifique la lista mediante el borrado de todos los elementos de la lista que tengan este valor. (8 puntos)
- 2. Construir un método *cantVocales* que determine la cantidad de vocales almacenadas en una lista de caracteres. (6 puntos)
- 3. Construir un método *imprimelnverso* que imprima los elementos de una lista enlazada de enteros en orden inverso a partir de una posición p. (6 puntos)

Lista de Cotejo

Criterio de Evaluación	Comprensión	Diseño del	Implementació	Manejo de Casos	Claridad y	Puntuación
	del Problema	Algoritmo (El	n del Método (El	Borde/Especiale	Legibilidad del	del Ejercicio
	(El método	algoritmo	método está	s (Se consideran	Código (El	
	aborda	diseñado es	implementado	y manejan	código es fácil	
	correctamente	lógico y	correctamente)	correctamente	de entender,	
	el problema	eficiente		casos como lista	con nombres	
	planteado.)	para la		vacía, elemento	de variables	
		tarea.)		no encontrado	significativos	
				(ejercicio 1), lista	y una	
				de caracteres	estructura	
				vacía (ejercicio 2),	clara.)	
				posición inválida		
				(ejercicio 3).		
Ejercicio 1: Borrar	Sí (1)	Sí (2)	Sí (3)	Sí (1)	Sí (1)	/8
Elementos (8 puntos)						
Ejercicio 2: Cantidad	No (0)	No (0)	No (0)	No (0)	No (0)	/6
de Vocales (6 puntos)						
Ejercicio 3: Imprimir	Sí (1)	Sí (1.5)	Sí (2)	Sí (0.5)	Sí (1)	/6
Inverso (6 puntos)		, ,	· ,			
TOTAL						
TOTAL						

Criterio de Evaluación	Comprensión del Problema (El método aborda correctamente el problema planteado.)	Diseño del Algoritmo (El algoritmo diseñado es lógico y eficiente para la tarea.)	Implementación del Método (El método está implementado correctamente)	Manejo de Casos Borde/Especiales (Se consideran y manejan correctamente casos como lista vacía, elemento no encontrado (ejercicio 1), lista de caracteres vacía (ejercicio 2), posición inválida (ejercicio 3).	Claridad y Legibilidad del Código (El código es fácil de entender, con nombres de variables significativos y una estructura clara.)	Puntuación del Ejercicio
Ejercicio 1: Borrar Elementos (8 puntos)	Sí (1)	Sí (2)	Sí (3)	Sí (1)	Sí (1)	/8
Ejercicio 2: Cantidad de Vocales (6 puntos)	No (0)	No (0)	No (0)	No (0)	No (0)	/6
Ejercicio 3: Imprimir Inverso (6 puntos)	Sí (1)	Sí (1.5)	Sí (2)	Sí (0.5)	Sí (1)	/6
TOTAL						