|  |  |  |
| --- | --- | --- |
| Logotipo, nombre de la empresa  El contenido generado por IA puede ser incorrecto. | Imagen que contiene Logotipo  El contenido generado por IA puede ser incorrecto. | Diagrama  El contenido generado por IA puede ser incorrecto. |

**INSTITUTO TECNOLÓGICO DE FRONTERA COMALAPA**

**MATERIA**

**ESTRUCTURA DE DATOS**

**TEMA**

**SINTRESIS**

**ESTUDIANTE**

**ISAIAS SALATHIEL LOPEZ TORRES**

**TERCER SEMESTRE, ING. SISTEMAS COMPUTACIONALES.**

**NC:241260085**

**MODALIDAD**

**ESCOLARIZADA**

**DOCENTE**

**ING. FRANCISCO JAVIER MINGO VELAZQUEZ**

**FRONTERA COMALAPA CHIAPAS, A 05 DE SEPTIEMBRE DEL 2025**

**INTRODUCCIÓN A LAS ESTRUCTURAS DE DATOS**

El estudio de las estructuras de datos constituye un pilar fundamental en el ámbito de la informática, ya que permite organizar, almacenar y manipular la información de manera eficiente, optimizando tanto el tiempo de ejecución como el uso de los recursos de memoria.

En primer lugar, la **clasificación de las estructuras de datos** distingue entre estructuras simples, que corresponden a los tipos primitivos como enteros, reales o caracteres, y las estructuras compuestas, que incluyen listas, pilas, colas, árboles y grafos. Asimismo, se diferencian según su administración en estructuras estáticas, de tamaño fijo, y dinámicas, que pueden modificarse durante la ejecución del programa.

Por otra parte, los **tipos de datos abstractos (TDA)** ofrecen un nivel de abstracción que permite definir las operaciones que se realizan sobre los datos sin necesidad de especificar su implementación. Entre los ejemplos más representativos de TDA se encuentran las pilas, colas, listas, árboles y grafos, los cuales poseen aplicaciones directas en la resolución de problemas computacionales y en el desarrollo de software.

El **manejo de memoria** resulta esencial para garantizar un uso óptimo de los recursos. En este sentido, la memoria estática se asigna en tiempo de compilación y se caracteriza por su tamaño fijo, mientras que la memoria dinámica se asigna en tiempo de ejecución, otorgando mayor flexibilidad y adaptabilidad a los programas.

Finalmente, el **análisis de algoritmos** permite evaluar el rendimiento de una solución en términos de tiempo y espacio. La **complejidad temporal** se centra en el número de pasos necesarios para ejecutar un algoritmo, mientras que la complejidad espacial analiza el consumo de memoria. La **eficiencia** de un algoritmo se alcanza al equilibrar ambos factores, lo que resulta clave en el diseño de programas robustos y escalables.

En conclusión, este tema subraya la importancia de comprender la relación entre estructuras de datos, memoria y algoritmos, ya que de ello depende el desarrollo de aplicaciones eficientes capaces de responder a las demandas actuales de procesamiento y gestión de información.