



# Estructura de Datos

Unidad I

Introducción a las Estructuras de Datos

# Estructura de datos<sup>(1)</sup>

- Se trata de un conjunto de variables de un determinado tipo agrupadas y organizadas de alguna manera para representar un comportamiento.
- Lo que se pretende con las estructuras de datos es facilitar un esquema lógico para manipular los datos en función del problema que haya que tratar y el algoritmo para resolverlo. En algunos casos la dificultad para resolver un problema radica en escoger la estructura de datos adecuada. Y, en general, la elección del algoritmo y de las estructuras de datos que manipulará estarán muy relacionadas.

# Estructura de datos<sup>(2)</sup>

- Una estructura de datos se refiere a la definición y organización de la estructura (espacio de memoria) donde se almacena el valor o valores que toma un dato en determinado instante.
- Cuando se habla de estructura de datos generalmente se asocia con los tipos de datos estructurados, por lo cual se dice que una estructura de datos es:
  - “La organización de espacios de memoria necesarios para almacenar una colección de valores de diferente o igual tipo de dato y que describe el comportamiento de los datos abstraídos de la realidad”.
- Una estructura de datos es referenciada a través de un identificador o nombre de variable.

# Clasificación de Estructura de datos

Existen tipos de datos simples y estructurados:

- Datos de tipo simple almacenan un solo valor a la vez.
- Datos de tipo estructurado almacenan una colección de valores de un mismo tipo de datos o de diferentes tipos.

Con respecto al manejo de memoria se clasifican en:

- Estáticas: su tamaño se define en tiempo de compilación y durante la ejecución del programa el tamaño de la estructura no cambia.
- Dinámicas: durante la ejecución del programa el tamaño de la estructura puede cambiar.

Con respecto a su organización se clasifican en:

- Lineales: se organizan como una lista secuencial de elementos (unidimensional). Ejemplo: pilas, colas, listas enlazadas, arreglos unidimensionales.
- No lineales: su organización es no lineal (multidimensional). Ejemplo: arreglos multidimensionales, árboles (organización jeraquica) y grafos.

# Tipos Abstractos de Datos (TAD)

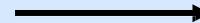
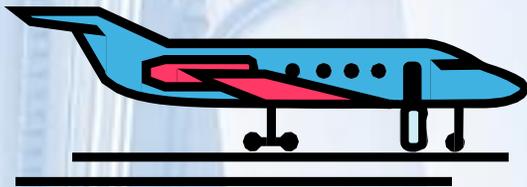
Abstracción: La abstracción es la capacidad para encapsular y aislar la información, del diseño y ejecución. Se refiere a ignorar los detalles no esenciales, tratando en su lugar con el modelo ideal del objeto y centrándonos en el estudio de sus aspectos esenciales.

Ejemplo: Una abstracción de un alumno para un sistema de inscripción sería:

- Propiedades: nombre, noctrl., carrera, semestre...
- Métodos: Carga\_materia, Eliminar\_materia, ....

# ¿Que es un TAD?

- Cuando se escribe un programa para resolver un problema, con el enfoque tradicional se pasa directamente de la realidad a una implementación en el lenguaje de programación.
- Los TAD establecen un nivel intermedio, donde se quiere moderar lo esencial de la realidad sin comprometerse con detalles de implementación. De hecho, es posible considerar diferentes implentaciones.



# Definición de un TAD

Un TAD es una estructura algebraica, o sea, un conjunto de objetos con ciertas operaciones definidas sobre ellos.

Ejemplos:

Una calculadora: los elementos que maneja son cantidades numéricas y las operaciones que tiene definidas sobre éstas son las operaciones aritméticas.

Una matriz: los elementos que maneja son las matrices y las operaciones son aquellas que nos permiten crearlas, sumarlas, invertirlas, etc.

# Elementos de un TAD

Con los TAD se identifican ciertas operaciones o partes del algoritmo que manipulan los datos:

Programa = Datos + Algoritmo de Datos + Algoritmos de Control

Algoritmo de datos: es la parte del algoritmo encargada de manipular las estructuras de datos del problema.

Algoritmo de control: es la parte restante, la que representa en sí el método de solución del problema, independiente hasta cierto punto de las estructuras de datos seleccionadas.

La Implementación de TAD son:

Datos + Algoritmo de Datos

El enfoque de desarrollo con TAD define programa como:

Implementación del TAD + Algoritmo de Control

# Tipo de operaciones en TAD

## *Creación de objetos:*

- Iniciales: Se utilizan para crear objetos del TAD, en cuya creación no se requieren ningunos objetos abstractos del mismo tipo.
- Constructores: Utilizadas para crear objetos del TAD cuya creación depende de objetos del mismo tipo.

## *Transformación de objetos:*

- Simplificadoras: Son operaciones cuyo codominio es el TAD que se define, pero que dan como resultado objetos que pueden ser descritos utilizando únicamente operaciones iniciales y constructoras.

## *Análisis de los elementos del TAD:*

- Analizadoras: Son operaciones cuyo codominio no es el TAD que se define, sino otro ya conocido. El propósito de este tipo de operaciones es obtener información concerniente a cualquiera de los objetos abstractos de tipo.

# Implementación de un TAD

- Un tipo abstracto de datos (TAD) es un tipo de dato definido por el programador que se puede manipular de un modo similar a los tipos de datos definidos por el sistema.
- Un tipo de dato abstracto corresponde a un conjunto (puede ser de tamaño indefinido) de valores legales de datos y un número de operaciones primitivas que se pueden realizar sobre esos valores.
- Los usuarios pueden crear variables con valores que están en el rango de valores legales y pueden operar sobre esos valores utilizando las operaciones definidas.
- Escoger unas estructuras de datos para representar cada uno de los objetos abstractos y escribir una rutina (Procedimiento o función) en un lenguaje de programación, que simule el funcionamiento de cada una de las operaciones de acuerdo con su especificación.