

# *Tipo Abstracto de Dato*

Programación digital II  
Escuela de Sistemas  
Facultad de Ingeniería  
Profesor: Gilberto Diaz

Con la aparición de los lenguajes de programación en los años 60 se definieron los **tipos de datos**  
Entendamos antes algunos conceptos básicos

**Dato:** En el computador un dato siempre se resume a un conjunto de bits. Un conjunto de reglas sobre esos bits forman un tipo de dato.

Por ejemplo:  
un dato de tipo entero  
un dato de tipo caracter.

**Valor:** Es un elemento perteneciente a un conjunto. Este conjunto representa un tipo de dato y todos sus elementos (valores) deben tener las mismas propiedades

El conjunto define las operaciones que se pueden aplicar a sus valores

Por ejemplo: El valor 2 es un elemento del conjunto siguiente

$$N = \{ 1, 2, 3 \dots \}$$

**Tipo de Dato:** Es la agrupación de un conjunto de valores sobre el cual se puede realizar un conjunto de operaciones. Ejemplo

El tipo de dato fecha podría estar representado por los atributos DIA, MES y AÑO, siendo los tres de tipo entero. Las operaciones aplicables sobre el tipo de dato fecha podrían ser:

- Mostrar fecha
- Incrementar día
- Incrementar mes
- Incrementar año
- Intervalo entre dos fechas

Los tipos de datos sirven para clasificar los objetos de los programas tales como:

- Variables
- Constantes
- Parámetros

Determinan los valores que pueden tomar estos objetos y las operaciones que se pueden realizar sobre ellos

Podemos resumir que un TAD es un conjunto de valores sobre los cuales se aplica un conjunto dado de operaciones que cumplen determinadas propiedades

En realidad, el concepto de TAD ya existe en los lenguajes de programación estructurados bajo la forma de los tipos predefinidos, que se pueden considerar como tipos abstractos con poco esfuerzo adicional.

Por ejemplo, consideremos el tipo de datos de los enteros que ofrece el lenguaje C.

La definición del TAD correspondiente consiste en determinar:

- cuáles son sus valores: los números enteros dentro del intervalo [minint, maxint];
- cuáles son sus operaciones: la suma, la resta, el producto, y el cociente y el resto de la división
- cuáles son las propiedades que cumplen estas operaciones: hay muchas; por ejemplo:  
 $a+b = b+a$ ,  $a*0 = 0$ , etc.

# ¿Por qué "abstracto"?

---

---

El calificativo "abstracto" no significa "surrealista" sino que proviene de "abstracción", y responde al hecho de que los valores de un tipo pueden ser manipulados mediante sus operaciones si se saben las propiedades que éstas cumplen, sin que sea necesario ningún conocimiento ulterior sobre el tipo.

En concreto, su implementación en la máquina es absolutamente irrelevante.

# ¿Por qué "abstracto"?

---

---

En el caso de los enteros del lenguaje C, cualquier programa escrito en este lenguaje puede efectuar la operación

$$x + y$$

(siendo  $x$  e  $y$  dos variables enteras), con la certeza de que siempre calculará la suma de los enteros  $x$  e  $y$ , independientemente de su representación interna en la máquina que está ejecutando el programa

La especificación de un TAD consiste en establecer las propiedades que lo definen.

Una especificación debe tener las siguientes características:

- **Precisa**: tiene que decir lo realmente imprescindible
- **General**: adaptable a diferentes contextos

- **Legible:** debe servir como instrumento de comunicación entre el especificador y el usuario del tipo y entre el especificador y el implementador
- **No ambigüa:** debe evitar problemas posteriores de interpretación

La especificación de un TAD debe ser única y además debe definir completamente su comportamiento a cualquier usuario

La especificación de un TAD consiste en determinar

- Determinar una representación para los valores del tipo y
- En codificar las operaciones

Se hace utilizando un lenguaje de programación convencional.

Para que sea útil la implementación, esta debe ser:

- **Estructurada:** para facilitar su desarrollo
- **Eficiente:** para optimizar los recursos del computador
- **Legible:** para facilitar su modificación y mantenimiento

Especificar el TAD número complejo:

Nota: suponemos que la parte real e imaginaria de los número complejos son enteros para simplicidad del ejercicio

El TAD número complejo se define de la siguiente manera:

- Cuáles son sus valores: todos los números complejos de parte real e imaginaria enteras y entre los valores *minint* y *maxint*
- Cuáles son sus operaciones:
  - Sumar
  - Crear un número complejo a partir de dos enteros
  - Obtener la parte real y la imaginaria.

El TAD número complejo se define de la siguiente manera:

- Cuáles son las propiedades sus operaciones: