



Materia: Estructura de Datos I  
INF 220

## “Tema 1. Introducción a las Estructuras de Datos”

**Docente: Ing. Ubaldo Pérez Ferreira**

**E-mail: [ubaperez@gmail.com](mailto:ubaperez@gmail.com)**

Santa Cruz de la Sierra – Bolivia © 2021

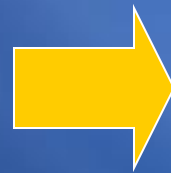


```
<?xml version="1.0" encoding="UTF-8" ?>
<wml>
  <card title="TV Interface">
    <channel="channel1.wml"> Bienvenidos a Wireless Tuna </a>
    <a href="uni.wml"> con el balde de 2003 </a>
    <a href="back.wml"> back </a>
  </card>
</wml>
```

# Introducción a las Estructuras de Datos

En el campo de la ciencia de la computación, hallamos la definición como el “estudio de algoritmos”, sin embargo, no se hace referencia a la palabra “dato”.

Si el computador es considerado como un medio y su fin.



El algoritmo es el medio, pero la transformación de datos es su objetivo.

Es muy frecuente definir un computador como una maquina que procesa datos, es decir, datos crudos son ingresados y se utiliza un algoritmo para transformarlo en datos refinados



```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- This is a WML document -->
<wml>
  <card title="TV Interface">
    <chref="channel1.wml"> B1: Canal Wireless Time </a>
    <a href="ani.wml"> ani con o bal des: 2003 </a>
    <a href="back.wml"> back </a>
  </card>
</wml>
```

# Introducción a las Estructuras de Datos

Por lo tanto, alternativamente diremos que la Ciencia de la Computación es el estudio de “**dato**”.

- i. Maquinas que almacenan datos.
- ii. Fundamento que describen que clase de datos son producidos a partir de datos crudos.
- iii. Representación de Estructuras de Datos.



```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<word>
  <card title="TV Interface">
    <a href="#channel1.xml">Bit Cerco Wireless Tune</a>
    <a href="#ami.xml">ami con o bal des: 2003</a>
    <a href="#back.xml">back</a>
  </card>
</word>
```

# Introducción a las Estructuras de Datos

Algoritmo

+

Estructura  
de Datos

=

Programa

En Programación Orientada a Objetos.

Objeto

Datos

Funciones

Encapsulamiento  
de Datos



```
<?xml version="1.0" encoding="UTF-8" ?>
<wml>
  <card title="TV Interface">
    <par>
      <a href="#">TV Interface</a>
      <a href="#">TV Interface</a>
    </par>
  </card>
</wml>
```

# Introducción a las Estructuras de Datos

Es así que, la Ciencia de la Computación puede ser definida como:

# El estudio del dato, su representación y transformación mediante computadores digitales

## Esto implica conocer:

# Las diferentes Estructuras de Datos.





# Introducción a las Estructuras de Datos

## Esto implica dominar dos técnicas:

- Habilidad para representar las diferentes formas de datos.
- Habilidad para analizar los algoritmos que operan sobre los datos.







# Tipo Abstracto de Datos. (TAD)

- Un TAD es definido por el programador.
- Están conformado por: Datos (Estructura de Datos) y Operaciones (Funciones y Procedimientos) que se realizan sobre esos datos.
- El conjunto de Operaciones debe ser Cerrado, es decir solo se debe acceder a los datos mediante estas operaciones.

```

<!--
  Copyright 2005 by Robert O. Orend
  All rights reserved.

  Permission is granted to copy, distribute, and/or modify this
  document under the terms of the Creative Commons Attribution
  License, which is available at http://creativecommons.org/licenses/by/2.0/
  or http://creativecommons.org/licenses/by/2.0/deed.pt_BR.

  In copying and distributing this document, you are encouraged to
  copy only those sections which you wish to copy.  However,
  entire documents are permitted to be copied under the following
  conditions:
  1. The copyright notice for this document must be preserved.
  2. The license notice for this document must be preserved.
  3. The document must be distributed under the Creative Commons
  Attribution License.

  This document is made available under the Creative Commons
  Attribution License.  For more information, see
  http://creativecommons.org/licenses/by/2.0/
-->
<?xml version="1.0" encoding="UTF-8" ?>
<channel name="wml">
  <title>TV Interface</title>
  <cat>TV</cat>
  <desc>TV Interface</desc>
  <url>http://www.orend.com</url>
  <image>http://www.orend.com</image>
  <logo>http://www.orend.com</logo>
  <copyright>2005 Robert O. Orend</copyright>
  <license>http://creativecommons.org/licenses/by/2.0/
  or http://creativecommons.org/licenses/by/2.0/deed.pt_BR</license>
  <item>
    <title>TV Interface</title>
    <cat>TV</cat>
    <desc>TV Interface</desc>
    <url>http://www.orend.com</url>
    <image>http://www.orend.com</image>
    <logo>http://www.orend.com</logo>
    <copyright>2005 Robert O. Orend</copyright>
    <license>http://creativecommons.org/licenses/by/2.0/
    or http://creativecommons.org/licenses/by/2.0/deed.pt_BR</license>
  </item>
</channel>
<card title="TV Interface">
  <cat>TV</cat>
  <desc>TV Interface</desc>
  <url>http://www.orend.com</url>
  <image>http://www.orend.com</image>
  <logo>http://www.orend.com</logo>
  <copyright>2005 Robert O. Orend</copyright>
  <license>http://creativecommons.org/licenses/by/2.0/
  or http://creativecommons.org/licenses/by/2.0/deed.pt_BR</license>
</card>
</wml>

```



# Tipo Abstracto de Datos. (TAD)

## Un TAD se compone de dos partes

## Interfaz. Se declaran los Datos y sus operaciones

**Implementación.** Contiene el código fuente de las operaciones y lo mantiene oculto al usuario.

[illegible]



# Tipo Abstracto de Datos. (TAD)

## Construcción de un TAD

La construcción de un TAD consta de dos fases bien diferenciadas entre ellas:

- la especificación (formal e informal)
- implementación.

Las características de un TAD no deben depender de su realización concreta, sino solamente de cómo queremos que sea su comportamiento, lo cual llamamos especificación.



```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<wml>
  <card title="TV Interface">
    <a href="#channel1.wml">Bit Cerco Wireless Tune</a>
    <a href="#ami.wml">ami con o bal des: 2003</a>
    <a href="#back.wml">back</a>
  </card>
</wml>
```

## Especificaciones informales.

La descripción de una abstracción se realiza con un lenguaje natural (ambiguas e imprecisas).

## Notación de especificación:

Partes que debe tener la especificación,  
Significado y tipo de cada parte,  
Partes que pueden ser opcionales, ...

```
<!--#include virtual="/includes/header.inc" -->
<!--#include virtual="/includes/footer.inc" -->
<!--#include virtual="/includes/nav.inc" -->
<!--#include virtual="/includes/body.inc" -->
</body>
</html>
```





## Ejemplo: concatenar dos cadenas.

- **Operación** concat (**ent** a,b: cadena; **sal** c: cadena)
- **Calcula**: la cadena de salida **c** es una nueva cadena que contiene los caracteres de **a** (en el mismo orden) seguidos de los caracteres de **b** (en el mismo orden).

```
<!--#include virtual="/includes/header.inc" -->
<!--#include virtual="/includes/footer.inc" -->
<!--#include virtual="/includes/nav.inc" -->
<!--#include virtual="/includes/body.inc" -->
</body>
</html>
```



## Especificaciones Formales - Tipo Abstracto de Datos (TAD).

## Notación

La descripción formal constará de cuatro partes:

**NOMBRE.** Nombre genérico del TAD.

**CONJUNTOS.** Conjuntos de datos que intervienen en la definición.

**SINTAXIS.** Signatura de las operaciones definidas.

$$\langle \text{nombre\_operación} \rangle : \langle \text{conj\_dominio} \rangle \rightarrow \langle \text{conj\_resultado} \rangle$$

**SEMÁNTICA.** Indica el significado de las operaciones.



# Especificaciones Formal – TAD Natural (Números Naturales)

NOMBRE natural (desde 0 hasta n)

CONJUNTOS N conjunto de naturales, B conjunto de valores booleanos

## SINTAXIS

1. cero:  $\rightarrow N$
2. sucesor( N )  $\rightarrow N$
3. escero( N )  $\rightarrow B$
4. igual( N , N )  $\rightarrow B$
5. suma( N , N )  $\rightarrow N$
6. EsPar(N)  $\rightarrow B$

} Funciones

SEMANTICA  $\forall m, n \in N$

6. escero (cero) = true
7. escero (sucesor (n)) = false
8. igual (cero, n) = escero (n)
9. igual (sucesor (n), cero) = false
10. igual (sucesor (n), sucesor (m)) = igual (n, m)
11. suma (cero, n) = n
12. suma (sucesor (m), n) = sucesor (suma (m, n))

} Axiomas



## Especificaciones Formal – TAD Natural Números Naturales

$\forall m, n \in \mathbb{N}$ , cual es el resultado de aplicar las siguientes expresiones.

1.  $\text{escero}(\text{cero}) = \text{escero}(\text{sucesor}(n)) = ?$
2.  $\text{suma}(\text{cero}, n) = ?$
3.  $\text{sucesor}(\text{sucesor}(\text{sucesor}(\text{cero}))) = ?$
4.  $\text{igual}(\text{sucesor}(n), \text{cero}) = ?$
5.  $\text{suma}(\text{sucesor}(n), \text{cero}) = ?$

## Que funciones y axiomas podemos adicionar al TAD Natural?

## Ejercicios, hacer la Especificación Informal y Formal para TAD Cadena

[illegible]

## Implementación – TAD Natural Números Naturales

Implementación del TAD Natural en el Lenguaje de Programación C#, definiendo la clase **clsNatu** dentro del Proyecto de Clases denominado **cApp**

```
namespace cApp ← Proyecto de Clases
{
    public class clsNatu ← Nombre de la clase del TAD Naturales
    {
        public int Cero()
        public bool esCero(int x)
        public int Sucesor(int x)
        public bool Igual(int x, int y)
        public int Suma(int a, int b)
    }
}
```

← Funciones





# Implementación – TAD Natural Números Naturales

## Detalle de las Funciones de la clase **clsNatu**.

```
public int Cero()
{
    return 0;
}
public bool esCero(int x)
{
    if (x==0)
    {
        return true;
    }
    return false;
}
public int Sucesor(int x)
{
    x = x + 1;
    return x;
}
```

```
public bool Igual(int x, int y)
{
    if (EsCero(x) == true || EsCero(y) == false)
        return false;

    if (EsCero(x) == false || EsCero(y) == true)
        return false;

    if (x == y)
        return true;
    return false;
}
```

Axionas/  
Semantica



```
<?xml version="1.0" encoding="utf-8" ?>
<wml>
<card title="TV Interface">
<chtml>
<channel>
<a href="#">TV Interface</a>
</channel>
</chtml>
</card>
</wml>
```