



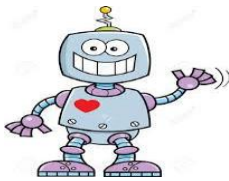
Facultad de Ciencias  
de la Administración

*Licenciatura en Sistemas*



## Algoritmos y Programación

### Atributos - Ligaduras



**Clase 00**



## Algoritmos y Programación

### ENTIDADES

Todo software está conformado por entidades, tales como variables, constantes, subprogramas, sentencias, expresiones, y otros.

### ATRIBUTOS

Son las propiedades o características de una entidad.

Cada entidad de un programa, está caracterizada por atributos.

**Ejemplo de atributos de una variable:**

- identificador o nombre
- tipo
- tiempo de vida
- valor
- etc.



¿ Quién ?  
¿ Cuándo ?



## Algoritmos y Programación

### Ligadura:

Asociar o vincular un atributo a una entidad se denomina "ligar" una entidad.

### Tiempo de Ligadura:

El momento en que ocurre una ligadura se denomina **«tiempo de ligadura»**.

El tiempo de ligadura es un concepto importante porque de él depende ciertas características del lenguaje de programación.

Existen distintos momentos en función del diseño del lenguaje de programación y de su implementación.



## Algoritmos y Programación

### Tiempo de Ligadura:

Se puede distinguir:

#### - **Durante la Ejecución** (*ligadura dinámica*).

Muchas ligaduras se establecen recién en la ejecución y pueden modificarse. Estas ligaduras provocan cierto costo en la ejecución pero son las que le dan flexibilidad al lenguaje.

#### - **Antes de la ejecución** (*ligadura estática*).

Si todas las ligaduras pudieran resolverse antes de la ejecución, el efecto de un programa podría anticiparse y entonces su ejecución será menos flexible pero mas segura.

- (*generalmente se liga con una declaración*).



## Un ejercicio básico

### Recordemos este simple ejemplo:

Ingresar dos números enteros por teclado, luego sumarlos e imprimir el resultado.

#### Inicio

Ingresar N1, N2

RES := N1 + N2

Imprimir RES

#### Fin



## Algoritmos y Programación

Tiempo de Ligadura:



Estática

### Ejemplo en Pascal

Las variables N1, N2 Y RES se declaran antes del inicio del programa (Begin)

Program Suma

Var

N1, N2, RES : integer;

Begin

writeln ("Ingrese 1er. Nro.");

ReadLn (N1);

writeln ("Ingrese 2er. Nro.");

ReadLn (N2);

RES := N1 + N2;

WriteLn ("Resultado: ", RES);

End.



Aquí se produce la ligadura

Se asocia a un pool de memoria y permanece ligado durante toda la ejecución del programa



## Algoritmos y Programación

Tiempo de Ligadura:



Dinámica



### Ejemplo en Python

Las variables N1, N2 Y RES se declaran en el mismo momento en que se las necesitan durante la ejecución del programa (no antes).

```
print ("Ingrese 1er. Nro.")
N1 = int(input())
print ("Ingrese 2do. Nro.")
N2 = int(input())
RES = int(N1 + N2)
print ("Resultado: ", RES)
```

Aquí se producen las ligadura



## Algoritmos y Programación

Tiempo de Ligadura:



Dinámica

### Ejemplo en Python

**PERO** también contempla otras variantes donde la flexibilidad es aún mayor.  
(El tipo es asumido por inferencia).

```
Valor = 12345
N1 = Valor + 1
print ("Valor de N1:", N1)
Valor = "Algoritmos"
print (Valor)
```

Aquí Python asume que la variable «Valor» es de tipo numérica entera.

Aquí Python cambia el tipo de «Valor» y asume que es string.



## Algoritmos y Programación

### Seguridad vs. Flexibilidad

La **flexibilidad compromete la seguridad** del lenguaje.

Dependiendo del dominio de aplicación del lenguaje de programación, será cual de las dos características termina siendo la más preponderante o buscada por el sistema.

Los lenguajes orientados al desarrollo de aplicaciones complejas y críticas, buscan mayor seguridad y se caracterizan por tener un sistema de tipos estricto (**estático**).

Los lenguajes script orientados al desarrollo rápido, buscan mayor flexibilidad de modo que se caracterizan por tener un sistema de tipos relajado (**dinámico**).



## Algoritmos y Programación

### Conclusiones

#### ¿ Qué enfoque utilizaremos nosotros ?

Para la codificación de nuestro pseudocódigo, proponemos utilizar el sistema de declarar las variables y ligar su nombre y tipo antes de la ejecución del Programa (en la sección «Var»).

**Luego:** siguiendo la estructura de un lenguaje académico que posteriormente utilizaremos, sugerimos trabajar siguiendo los ejemplos que detallamos en las diapositivas de nuestras clases.

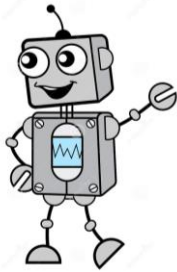


Facultad de Ciencias  
de la **Administración**

**Licenciatura en Sistemas**



## *Algoritmos y Programación*



**Atributos – Ligaduras**

**FIN**  
**Clase 00**

Prof. Miguel Fernández