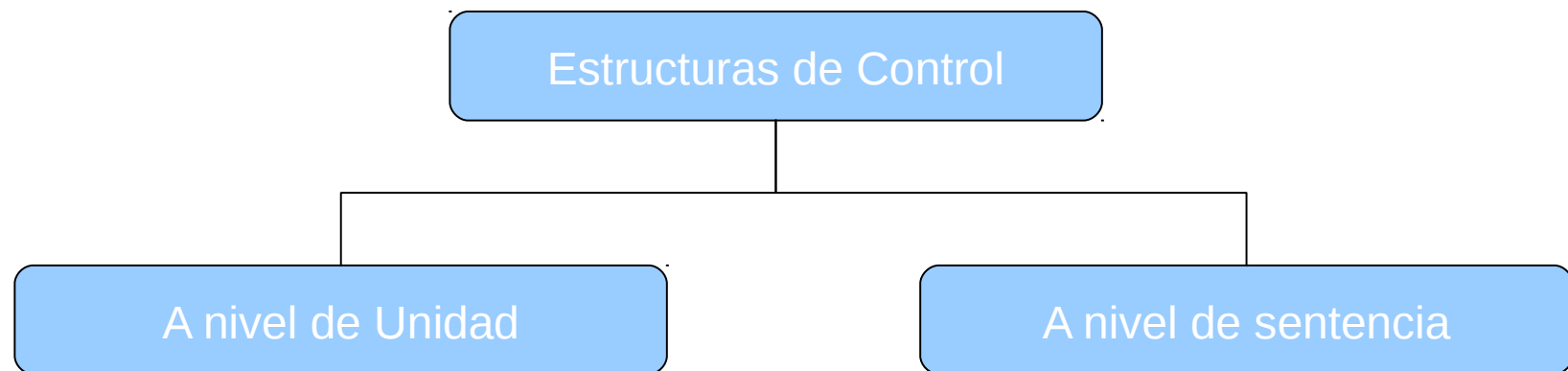


# Conceptos de Lenguajes de Programación

Estructuras de control

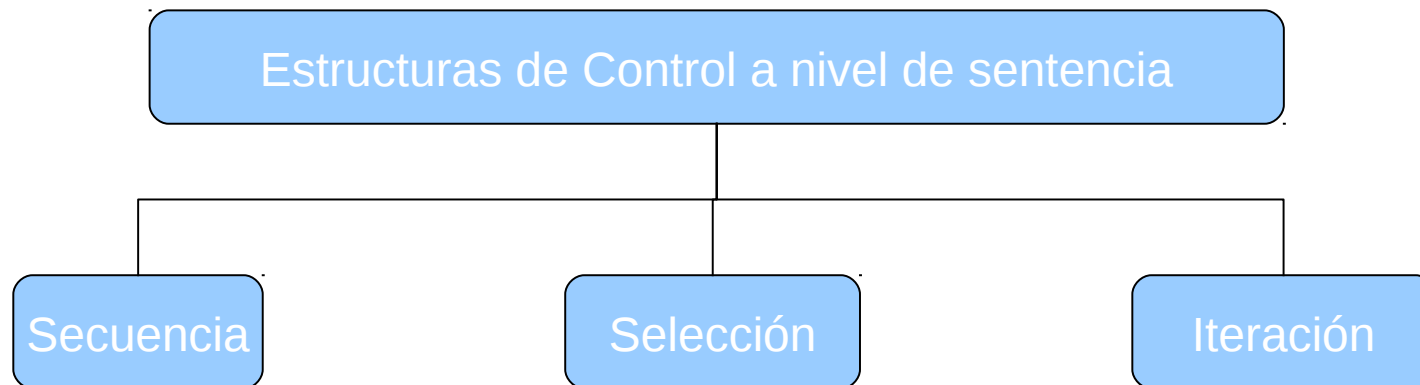
# Estructuras de Control

- Son el medio por el cual los programadores pueden determinar el flujo de ejecución entre los componentes de un programa



**A Nivel de Unidad :** Cuando el flujo de control se pasa entre unidades. Intervienen los pasajes de parámetros.

**A Nivel de Sentencia:** Se dividen en tres grupos



# Secuencia

Es el flujo de control más simple.

Indica la ejecución de una sentencia a continuación de otra.

El delimitador más general y más usado es el “ ; ”.

Hay lenguajes que no tienen delimitador y establecen que por cada línea vaya una instrucción.

Se los llaman **orientados a línea**. Ej: Fortran

Se pueden **agrupar** sentencias en una, llamada **Sentencia compuesta**, llevan delimitadores como **Begin** y **End**. Ej: Algol, Pascal, { y } en C, C++, etc.

# Distinción entre sentencia y expresión

En cualquier lenguaje convencional, ej. Pascal, existe diferencia entre **sentencia de asignación** y **expresión**

En otros lenguajes tales como **C** definen la **sentencia de asignación**, como una **expresión con efectos laterales**.

Las sentencias de asignación **devuelven valores**.

Evalúa de derecha a izquierda

Ejemplo **a=b=c=0;**

La mayoría de los lenguajes de programación requieren que sobre el lado izquierdo de la asignación aparezca un l-valor. C permite **cualquier expresión** que denote un l-valor.

Ej.:  $(i < j ? x : y) = 5$

# Selección

- Esta estructura de control permite que el programador pueda expresar una elección entre un cierto número posible de sentencias alternativa

## ► If lógico de Fortran:

**If** (condición lógica) sentencia

Si la condición es verdadera ejecuta la sentencia

## Condición:

- circuito corto:  $(2 > 1 \ \& \ \text{valor}) \rightarrow$  al comprobar que parte de la condición es falsa, no continúa con la comprobación y el resultado será falso. Esto indica que si valor es un llamado a una función, esta nunca será ejecutada.

# Selección – Selección múltiple

Sentencia de selección entre dos o más opciones

**Select**

**when(A) sentencia1;**

**when(B) sentencia2**

**.....**

**Otherwise sentencia n;**

**End;**

A considerar:

¿Qué tipo de condiciones se dan?

¿De qué manera se implementa la selección según el lenguaje?

¿Sentencia break, para qué?

¿Maneja sentencias por defecto (default – otherwise? ¿Cómo?

# Iteración

Este tipo de instrucciones se utilizan para representar aquellas acciones que se repiten un cierto número de veces

## ► Sentencia For de Java

```
for (int i =0; i<10; i++){  
    .....  
}
```

A considerar:

- ¿Es posible modificar la variable de iteración?
- ¿Es posible modificar los límites de iteración?
- ¿Cómo lo implementa cada lenguaje?



# Iteración – otras estructuras

## Otras sentencias iterativas: while y until

**While:** Estructura que permite repetir un proceso mientras se cumpla una condición. La condición se evalúa antes de que se entre al proceso.

**Until:** Estructura que permite repetir un proceso mientras se cumpla una condición. La condición se evalúa al final del proceso, por lo que por lo menos una vez el proceso se realiza.