

## Laboratorio 2

### Construcción de Compiladores

---

## Definición de Lenguaje Intermedio a generar en fase de compilación.

### Diseño de código intermedio:

La arquitectura propuesta para la implementación de código intermedio estará basada en la programación orientada a objetos y haremos un objeto que represente a los temporales.

#### objeto Temporal:

- **ID** public static int
  - Un valor único que representa a cada temporal.

#### objeto TemporalContainer:

- **stack** public temporal array
  - Una pila encargada de guardar todos los objetos Temporal.
- **def** addTemporal(params) public Temporal
  - Agrega un nuevo objeto Temporal.
- **def** deleteTemporal(ID) public void
  - Elimina un objeto Temporal por ID.
- **def** getTemporal(ID) public Temporal
  - Devuelve un objeto Temporal por ID.
- **def** addCode(params) public void
  - Agrega un nuevo código de tres direcciones.
- **def** deleteCode(ID) public void
  - Elimina un código de tres direcciones.

Se demostrará el código intermedio con código de tres direcciones. A partir de esto se definirán las siguientes reglas generales:

1.  $a = b \text{ operador } c$ 
  - Operador puede ser binario aritmético o una expresión lógica.
2.  $a = b$ 
  - Una asignación de un valor a otro, como un overwrite.
3.  $a = \text{operador } b$ 
  - Operadores unitarios
4. *if condition do L*
  - *condition*: Es una operación o comprobación de valor cuyo resultado es un bool true y L es un set de operaciones.
5. *while condition reloop L*
  - reloop se refiere a que se hará múltiples veces L mientras *condition* sea true.

### Operaciones Básicas:

- Suma
  - $a = b + c + d$ 
    - $t1 = c + d$
    - $t2 = b + t1$
    - $a = t2$

Operador	Valor1	Valor2	Resultados
+	c	d	t1
+	b	t1	t2
=	t2		a

- Resta
  - $a = b - c - d$ 
    - $t1 = c - d$
    - $t2 = b - t1$
    - $a = t2$

Operador	Valor1	Valor2	Resultados
-	c	d	t1
-	b	t1	t2
=	t2		a

- Multiplicación

- $a = b * c + d$ 
  - $t1 = b * c$
  - $t2 = t1 + d$
  - $a = t2$

Operador	Valor1	Valor2	Resultados
*	b	c	t1
+	t1	d	t2
=	t2		a

- División

- $a = b / c * d$ 
  - $t1 = c * d$
  - $t2 = b / t1$
  - $a = t2$

Operador	Valor1	Valor2	Resultados
*	c	d	t1
/	b	t1	t2
=	t2		a

## Muestra de tabla de símbolos, sobre un programa en YAPL, que muestre:

(Revisar código de github)

Tipo	Cantidad de bytes
Integer	4 bytes
String	10 bytes máximo
Boolean	1 byte