

# Sobrecarga de Operadores

4.2

Programación II y Laboratorio de Computación II

Edición 2018

# Características

- Sobrecargar un operador consiste en modificar su comportamiento cuando este se utiliza con una determinada clase.

```
[acceso] static TipoRetorno operator nombreOperador (Tipo a[, Tipo b])  
{  
    //...  
}
```

**Nota:** El modificador de acceso no podrá ser de un ámbito mayor que el de la clase.

# Operadores Sobrecargables

Operadores	Tipos
<b>+, -, !, ~, ++, --, true, false</b>	<b>Unarios.</b>
<b>+, -, *, /, %, &amp;,  , ^, &lt;&lt;, &gt;&gt;</b>	<b>Binarios.</b>
<b>==, !=, &lt;, &gt;, &lt;=, &gt;=</b>	<b>Comparación. (*)</b>

(\*) Nota: Los operadores de Comparación, si son sobrecargados, se deben sobrecargar en pares; es decir, si se sobrecarga el operador ==, se deberá sobrecargar el operador !=.



# Operadores No Sobrecargables

Operadores	Tipos
&&,	Condicionales Lógicos.
[]	Indexador de Array. (*)
()	Casting. (**)
+=, -=, *=, /=, %=, &=,  =, ^=, <<=, >>=	Asignación. (***)
=, ., ?.:, ->, new, is, sizeof, typeof	Estos operadores no se pueden sobrecargar.

# Operadores No Sobrecargables

- (\*) Indexador de Array:
  - Se pueden definir indexadores.
- (\*\*) Casting:
  - Se pueden definir nuevos operadores de conversión.
- (\*\*\*) Asignación:
  - El operador `+=`, por ejemplo, es evaluado usando el operador `+`, el cual puede ser sobrecargado.

# Operadores de Conversión

- Las conversiones definidas permiten hacer compatibles tipos que antes no lo eran.
- Los operadores de conversión pueden ser implícitos o explícitos.
- Los operadores de conversión explícitos son muy usados cuando se quiere que los usuarios estén conscientes que una conversión se llevará a cabo.



# Sintaxis

- Implícitos

```
[acceso] static implicit operator nombreTipo(Tipo a)
{
    //...
}
```

- Explícitos

```
[acceso] static explicit operator nombreTipo(Tipo a)
{
    //...
}
```