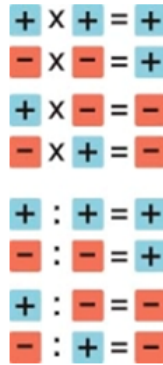


## 5.C. Métodos.

### 1. Métodos.

#### 1.8. Sobrecarga de operadores.

Del mismo modo que hemos visto la posibilidad de sobrecargar métodos (disponer de varias versiones de un método con el mismo nombre cambiando su lista de parámetros), podría plantearse también la opción de **sobrecargar operadores del lenguaje tales como +, -, \*, <, >, etc.** para **darles otro significado dependiendo del tipo de objetos con los que vaya a operar.**



En algunos casos puede resultar útil para ayudar a mejorar la legibilidad del código, pues esos operadores resultan muy intuitivos y pueden dar una idea rápida de cuál es su funcionamiento.

Un típico ejemplo podría ser el de la sobrecarga de operadores aritméticos como la suma (+) o el producto (\*) para operar con fracciones. Si se definen objetos de una clase **Fracción** (que contendrá los atributos numerador y denominador) podrían sobrecargarse los operadores aritméticos (habría que redefinir el operador suma (+) para la suma, el operador asterisco (\*) para el producto, etc.) para esta clase y así podrían utilizarse para sumar o multiplicar objetos de tipo **Fracción** mediante el algoritmo específico de suma o de producto del objeto **Fracción** (pues esos operadores no están preparados en el lenguaje para operar con esos objetos).

En algunos lenguajes de programación como por ejemplo C++ o C# se permite la sobrecarga, pero no es algo soportado en todos los lenguajes. ¿Qué sucede en el caso concreto de Java?

**El lenguaje Java no soporta la sobrecarga de operadores.**

En el ejemplo anterior de los objetos de tipo **Fracción**, habrá que declarar métodos en la clase **Fracción** que se encarguen de realizar esas operaciones, pero no lo podremos hacer sobrecargando los operadores del lenguaje (los símbolos de la suma, resta, producto, etc.). Por ejemplo:

```
public Fraccion sumar (Fraccion sumando)
```

```
public Fraccion multiplicar (Fraccion multiplicando)
```

Y así sucesivamente...

Dado que en este módulo se está utilizando el lenguaje Java para aprender a programar, no podremos hacer uso de esta funcionalidad. Más adelante, cuando aprendas a programar en otros lenguajes, es posible que sí tengas la posibilidad de utilizar este recurso.

#### Autoevaluación

La sobrecarga de operadores en Java permite "rescribir" el significado de operadores del lenguaje tales como +, -, \*, <, >, etc. Esto puede resultar muy útil a la hora de mejorar la legibilidad del código cuando definimos por ejemplo nuevos objetos matemáticos (números racionales, números complejos, conjuntos, etc.). ¿Verdadero o falso?

- ☐ Verdadero.  
☐ Falso.