

# Programación II

## Tema 4

jtorresmat@nebrija.es

# Índice

- Sobrecarga
  - Métodos
  - Operadores
- Ejemplos

# Sobrecarga de métodos

- Con las funciones podíamos sobrecargar:
  - Varias funciones con la misma etiqueta
  - Que difieren en:
    - Tipo de argumentos
    - Número de argumentos
- Con los métodos podemos hacer lo mismo, ya que son funciones encapsuladas en una clase

# Sobrecarga de métodos

- Ejemplo sobrecarga de métodos.

# Índice

- Sobrecarga
  - Métodos
  - Operadores
- Ejemplos

# Sobrecarga de operadores

- Operador: símbolo que indica cómo se van a manipular los operandos.
- Operando/s y operador forman una expresión.
- Existe una jerarquía (prioridad) que se puede alterar con el uso de paréntesis

$$2+3*8 \neq (2+3)*8$$

[https://en.cppreference.com/w/cpp/language/operator\\_precedence](https://en.cppreference.com/w/cpp/language/operator_precedence)

# Sobrecarga de operadores

- La sobrecarga de operadores nos permite ampliar sus capacidades en P00.
- La declaración y definición es parecida a la de una función.
- Sintaxis:

`"tipo" operator "símbolo" (operadores) ;`  
valor de retorno      palabra reservada      operador      los que necesite la operación...



# Sobrecarga de operadores

- Ejemplo: programa que declare 2 tipos enteros A y B, los sume y muestre por pantalla el resultado.

# Sobrecarga de operadores

- Ejemplo: en el mismo programa anterior crear una estructura para albergar un número complejo. Cambiar el tipo de dato por el de nuestra estructura "nComplejo"

Compilar -> ¿Qué pasa?

El compilador no sabe como sumar estructuras nComplejo -> Vamos a enseñarle

# Sobrecarga de operadores

- En el ejemplo anterior hemos sobrecargado un operador fuera del main (parecido a una función). También se puede hacer dentro de una clase (parecido a un método).
- Sintaxis:

*"tipo" nombre clase:: **operator** "símbolo"(operadores);*

# Sobrecarga de operadores

- Ejemplo sobrecarga operador para una clase:

El ejemplo anterior de suma de números complejos, pero utilizando una clase

[código](#)

# Sobrecarga de operadores

- ¿Podemos sobrecargar todos los operadores?
  - No, C++ no permite sobrecarga para:
    - `#_##_._:_.::_.*_?:_sizeof_typeof`
- En esta lista no sale `>>` ni `<<` de `iostream`, entonces podemos sobrecargarlos...

Vamos a “terminar de arreglar” el ejemplo de los números complejos :)

Sobrecarga `>>` y `<<` [código](#)

# Sobrecarga de operadores

- Sobrecarga de operadores de flujo << y >>
  - Sintaxis flujo de salida <<

```
std::ostream& operator<<(std::ostream &os, Tipo const &a){  
    // lo que sea ...  
    return os;  
}
```
  - Sintaxis flujo de entrada >>

```
std::istream& operator>>(std::istream &os, Tipo &a){  
    // lo que sea ...  
    return is;  
}
```
  - Retornan una referencia para poder concatenar

# Sobrecarga de operadores

- Sobrecarga de operadores dentro de una clase:
  - Operadores unarios y binarios:
    - Binario: operador con dos operandos
      - Se declaran con un solo parámetro, ya que el primero se pasa como "this" (puntero al mismo objeto)
    - Unario: operador con un solo operando
      - Se declaran sin parámetros, ya que el único parámetro se pasa como "this"

# Sobrecarga de operadores

- En programación II nos vamos a “quedar” con:
  - Sobrecarga de operadores aritméticos (+; -; \*; /; %)
  - Sobrecarga operadores lógicos (<; <=; >; >=; ==; !=; &&; ||)
  - Sobrecarga operadores de flujo(<<; >>)
- Pero tened en cuenta que todo lo que no está en la lista de la diapositiva nº13 se puede sobrecargar en C++
  - [https://www.zator.com/Cpp/E4\\_9\\_18.htm](https://www.zator.com/Cpp/E4_9_18.htm)



# Sobrecarga de operadores

- Ejemplo operador unario:
  - Añadir incremento a clase Complejo, que sume 1 a la parte real y 1 a la imaginaria

[código](#)

# Sobrecarga de operadores

- Ejemplo operador binario: clase que tenga horas, minutos y segundos. Implementar un método mostrar y realizar sobrecarga del operador suma.

[código](#)

# Índice

- Sobrecarga de operadores
- Ejemplos

# Ejemplos

- Ejemplo: implementar una clase persona, con los atributos edad y nombre. Hacer un tipo de datos `std::set` que albergue personas de manera que no puedan repetirse personas con la misma edad.
  - error
  - ok

# Ejemplos

- Ejemplo: en el ejemplo de los números complejos con estructura implementar sobrecarga operadores suma, resta y multiplicación.

[código](#)

# Documentación en GitHub

- <https://github.com/Nebrija-Programacion/Programacion-II>
  - Sobrecarga operadores aritméticos:  
(<https://github.com/Nebrija-Programacion/Programacion-II/blob/master/temario/sobrecargaopar.md>)
  - Sobrecarga operadores lógicos:  
(<https://github.com/Nebrija-Programacion/Programacion-II/blob/master/temario/sobrecargaoplog.md>)
  - Sobrecarga operadores de flujo:  
(<https://github.com/Nebrija-Programacion/Programacion-II/blob/master/temario/sobrecargaopos.md>)