

# INFORMÁTICA INDUSTRIAL

## PROGRAMACIÓN BÁSICA C++

M. Abderrahim, A. Castro, J. C. Castillo  
Departamento de Ingeniería de Sistemas y Automática

**uc3m** | Universidad **Carlos III** de Madrid

# 3. Operadores matemáticos

# Operadores matemáticos

- Suma +
- Resta -
- Multiplicación \*
- División /
- Resto (enteros) %

# Operadores matemáticos

- Precedencia
  - Multiplicación, división y resto > suma y resta
  - Multiplicación = división=resto
  - suma = resta | multiplicación = división = resto
    - Criterio: Primera operación por la izquierda
  - Uso de paréntesis:
    - $5+4*3=5+12=17$
    - $(5+4)*3=9*3=27$

# Operadores matemáticos

- Precedencia y tipos

int i;

$i=5*32/9;$      $i=160/9=17$

$i=5/9*32;$      $i=0*32=0$

$i=5/9.0*32;$      $i=0.555*32=17$

# Operadores matemáticos

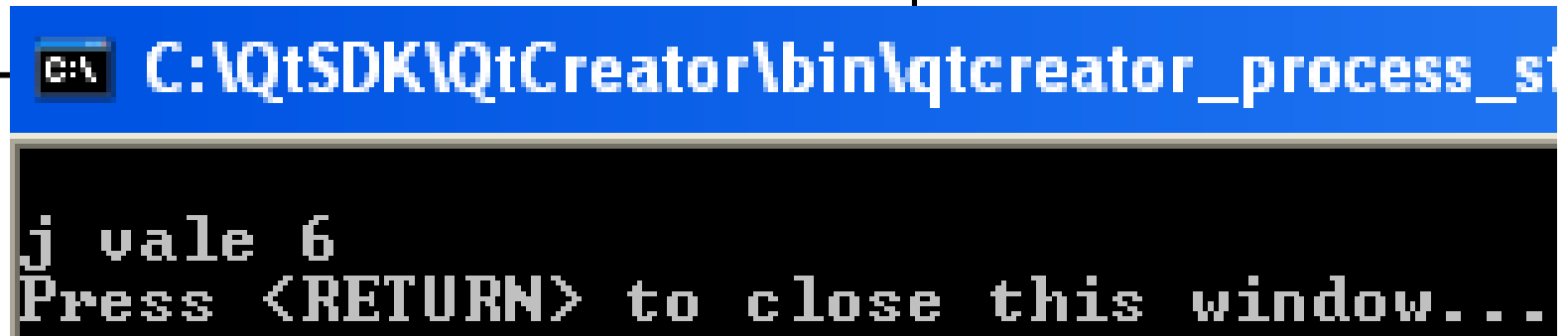
- Se pueden aplicar sobre constantes
- También se pueden aplicar sobre variables
- Se pueden combinar ambos.

# Operadores matemáticos

```
#include <iostream>

using namespace std;

int main()
{
    int i, j;
    i = 2;
    j = 3 * i;
    cout << "\n j vale " << j << endl;
    return 0;
}
```



The screenshot shows a console window with a blue title bar containing the path "C:\QtSDK\QtCreator\bin\qtcreator\_process\_st...". The console output displays "j vale 6" followed by the instruction "Press <RETURN> to close this window..."

# Operadores ++ --

++i es equivalente a  $i=i+1$

i++ es equivalente a  $i=i+1$

$(i+j)++$  *es ilegal*

--i es equivalente a  $i=i-1$

i-- es equivalente a  $i=i-1$

$(i+j)--$  *es ilegal*

- **En expresiones:**

++i hace primero la suma y luego asigna el valor

i++ asigna primero el valor y luego hace la suma



# Operadores ++ --

```
#include <iostream>
using namespace std;
/* Programa ejemplo */
void main() {
    int a,b,c;
    a=b=c=0;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
    a=++b + ++c;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
    a=b++ + c++;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
    a=b-- + --c;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
}
```

	a	b	c
a=b=c=0;	0	0	0
a=++b + ++c;	?	?	?

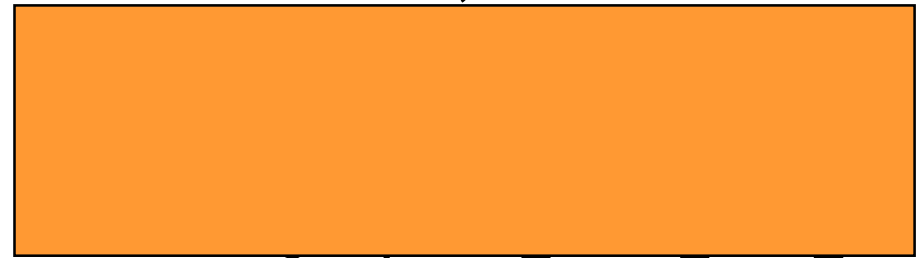
a=b++ + c++;	?	?	?
--------------	---	---	---

a=b-- + --c;	?	?	?
--------------	---	---	---

# Operadores ++ --

```
#include <iostream>
using namespace std;
/* Programa ejemplo */
void main() {
    int a,b,c;
    a=b=c=0;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
    a=++b + ++c;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
    a=b++ + c++;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
    a=b-- + --c;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
}
```

	a	b	c
a=b=c=0;	0	0	0
<hr/>			
a=++b + ++c;	?	?	?
++b	0	<b>1</b>	0
++c	0	1	<b>1</b>
a=b+c;	<b>2</b>	1	1
<hr/>			
a=b++ + c++;	?	?	?



a=b-- + --c; ? ? ?



# Operadores ++ --

```
#include <iostream>
using namespace std;
/* Programa ejemplo */
void main() {
    int a,b,c;
    a=b=c=0;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
    a=++b + ++c;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
    a=b++ + c++;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
    a=b-- + --c;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
}
```

	a	b	c
a=b=c=0;	0	0	0
<hr/>			
a=++b + ++c;	?	?	?
++b	0	<b>1</b>	0
++c	0	1	<b>1</b>
a=b+c;	<b>2</b>	1	1
<hr/>			
a=b++ + c++;	?	?	?
a=b+c	<b>2</b>	1	1
b++	2	<b>2</b>	1
c++;	2	2	<b>2</b>
<hr/>			
a=b-- + --c;	?	?	?



# Operadores ++ --

```
#include <iostream>
using namespace std;
/* Programa ejemplo */
void main() {
    int a,b,c;
    a=b=c=0;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
    a=++b + ++c;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
    a=b++ + c++;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
    a=b-- + --c;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
}
```

	a	b	c
a=b=c=0;	0	0	0
<hr/>			
a=++b + ++c;	?	?	?
++b	0	<b>1</b>	0
++c	0	1	<b>1</b>
a=b+c;	<b>2</b>	1	1
<hr/>			
a=b++ + c++;	?	?	?
a=b+c	<b>2</b>	1	1
b++	2	<b>2</b>	1
c++;	2	2	<b>2</b>
<hr/>			
a=b-- + --c;	?	?	?
--c	2	2	<b>1</b>
a=b+c	<b>3</b>	2	1
b--;	3	<b>1</b>	1

# Operadores ++ --

```
#include <iostream>
using namespace std;
/* Programa ejemplo */
void main() {
    int a,b,c;
    a=b=c=0;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
    a=++b + ++c;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
    a=b++ + c++;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
    a=b-- + --c;
    cout<<a<<"\t"<<b<<"\t"<<c<<endl;
}
```

	a	b	c
a=b=c=0;	0	0	0
<hr/>			
a=++b + ++c;	?	?	?
++b	0	<b>1</b>	0
++c	0	1	<b>1</b>
a=b+c;	<b>2</b>	1	1
<hr/>			
a=b++ + c++;	?	?	?
a=b+c	<b>2</b>	1	1
b++	2	<b>2</b>	1
c++;	2	2	<b>2</b>
<hr/>			
a=b-- + --c;	?	?	?
--c	2	2	<b>1</b>
a=b+c	<b>3</b>	2	1
b--;	3	<b>1</b>	1

# Operadores de asignación

- = += -= \*= /=
- *variable operador de asignación expresión*

k += 2

- *variable = variable operador expresión*

k = k + 2

- Ej.:

k\*=2

k\*=3+x



# Operadores de asignación

- = += -= \*= /=
- *variable operador de asignación expresión*

k += 2

- *variable = variable operador expresión*

k = k + 2

- Ej.:

k\*=2      k=k\*2

k\*=3+x    k=k\*(3+x)

# Operadores de asignación

- = += -= \*= /=
- *variable operador de asignación expresión*

k += 2

- *variable = variable operador expresión*

k = k + 2

- Ej.:

k\*=2      k=k\*2

k\*=3+x    k=k\*(3+x)



# Operaciones con bits

- Complemento a uno  $\sim$

$$\begin{array}{r} \sim \quad 00000101(5) \\ = \quad 11111010 \quad (250) \end{array}$$

- AND  $\&$

$$\begin{array}{r} 00000101(5) \\ \underline{00000110(6)} \\ \& \quad 00000100(4) \end{array}$$

- OR  $|$

$$\begin{array}{r} 00000101(5) \\ \underline{00000110(6)} \\ | \quad 00000111 \quad (7) \end{array}$$

- OR Exclusivo  $\wedge$

$$\begin{array}{r} 00000101(5) \\ \underline{00000110(6)} \\ \wedge \quad 00000011(3) \end{array}$$

# Operaciones con bits

- Desplazamientos

- variable *desplazamiento* num\_bits

i=5;

i=i << 2 (dos bits a la izda) | i=i >> 2 (dos bits a la dcha)

00000101 (5)

00000101 (5)

<< 2

>> 2



# Operaciones con bits

- Desplazamientos

- variable *desplazamiento* num\_bits

i=5;

i=i << 2 (dos bits a la izda) | i=i >> 2 (dos bits a la dcha)

00000101 (5)

00000101 (5)

<< 2

>> 2

= 00010100 (20)

= 00000001 (1)

# Operaciones con bits

- Desplazamientos

- variable *desplazamiento* num\_bits

i=5;

i=i << 2 (dos bits a la izda) | i=i >> 2 (dos bits a la dcha)

00000101 (5)

00000101 (5)

<< 2

>> 2

= 00010100 (20)

= 00000001 (1)

# INFORMÁTICA INDUSTRIAL

## PROGRAMACIÓN BÁSICA C++

M. Abderrahim, A. Castro, J. C. Castillo  
Departamento de Ingeniería de Sistemas y Automática

**uc3m** | Universidad **Carlos III** de Madrid