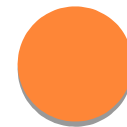


OPERADORS D'ASSIGNACIÓ

- La següent taula mostra els operadors de drecera(“atajo”) **d'assignació** i els seus equivalents largs:

Operador	Uso	Equivalent a
<code>+=</code>	<code>op1 += op2</code>	<code>op1 = op1 + op2</code>
<code>-=</code>	<code>op1 -= op2</code>	<code>op1 = op1 - op2</code>
<code>*=</code>	<code>op1 *= op2</code>	<code>op1 = op1 * op2</code>
<code>/=</code>	<code>op1 /= op2</code>	<code>op1 = op1 / op2</code>
<code>%=</code>	<code>op1 %= op2</code>	<code>op1 = op1 % op2</code>
<code>&=</code>	<code>op1 &= op2</code>	<code>op1 = op1 & op2</code>

- Exemple:
 - `int num = 5;`
 - `Num += 5;` `//num = 10, equival a num = num + 5`



OPERADORS UNARIS O UNITARIS

Operador	Uso	Operación
&	$op1 \& op2$	AND
	$op1 op2$	OR
^	$op1 \wedge op2$	OR Exclusivo
~	$\sim op2$	Complemento

- **Exemple : 12 & 13**
- El resultat d'aquesta operació es 12. Per què?.
- La representació en binario de 12 es **1100**,
- i de 13 es **1101**.
- La funció **AND** posca el bit de resultat a **1** si els dos bits dels operands son 1, sinó, el bit de resultat es 0:

```
  1101
& 1100
-----
  1100
```



OPERADORS UNARIS O UNITARIS

- Exemple: `int s1 = ~2;`
- **el complement a un de 2** es assignat al enter amb signe s1
- La representació binària dels complements a un dels decimals 0, 1 i 2 (els representem com un octet) son:

0	==	0000 0000	~ 0	==	1111 1111
1	==	0000 0001	~ 1	==	1111 1110
2	==	0000 0010	~ 2	==	1111 1101

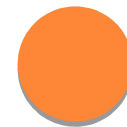
- Per tant, `s1 = 1111 1101`



OPERADORS DE DESPLAÇAMENT DE BITS

Operador	Uso	Operación
>>	<i>op1 >> op2</i>	Desplaza los bits de op1 a la derecha op2 veces
<<	<i>op1 << op2</i>	Desplaza los bits de op1 a la izquierda op2 veces
>>>	<i>op1 >>> op2</i>	Desplaza los bits de op1 a la derecha op2 veces (sin signo)

- Exemple: $13 \gg 1$;
- La representació en binari del número 13 es 1101.
- 1101 desplaçat una posició a la dreta és , 110 (6 en decimal).
- El bit més a la dreta es perd.
- Reomplim:
 - amb signe:
 - en positiu reomplim amb 0's
 - i en negatiu amb 1's.
 - Sense signe: sempre amb 0's.



OPERADORS DE DESPLAÇAMENT DE BITS

- $A \ll B$: Desplaçament a la esquerra de A, B bits reomplim amb **ceros** per la dreta.
- $A \gg B$: Desplaçament a la dreta de A, B bits reomplim amb el **bit de signe** per la esquerra.
- $A \ggg B$: Desplaçament a la dreta de A, B bits reomplim amb **ceros** per la equerra.
- **Exemple:**
- `int num=5;`
- `num = num << 1; //equival a num = num * 2;`
- `num = num >> 1; //equival a num = num / 2;`



PRECÈDENCIA DELS OPERADORS

- Primerament procedeixen els unaris, després els aritmètics, després els de bits, posteriorment els relacionals, darrere venen els booleans i per últim el operador d'assignació.

Precedencia de operadores

Descripción	Operadores
operadores posfijos	op++ op--
operadores unarios	++op --op +op -op ~ !
multiplicación y división	* / %
suma y resta	+ -
desplazamiento	<< >> >>>
operadores relacionales	< > <= >=
equivalencia	== !=
operador AND	&
operador XOR	^
operador OR	
AND booleano	&&
OR booleano	
condicional	?:
operadores de asignación	= += -= *= /= %= &= ^= = <<= >>= >>>=



○ Exercicis

