



UNIVERSITÀ DI PARMA

Le Espressioni

*We must either institute conventional forms
of expression or else pretend that we have
nothing to express.*

George Santayana, Soliloquies in England

- Operatori & Operandi
 - Tipologie
- Espressioni
- Valutazione
- Precedenza & Associatività
- Effetti collaterali

SUMMARY

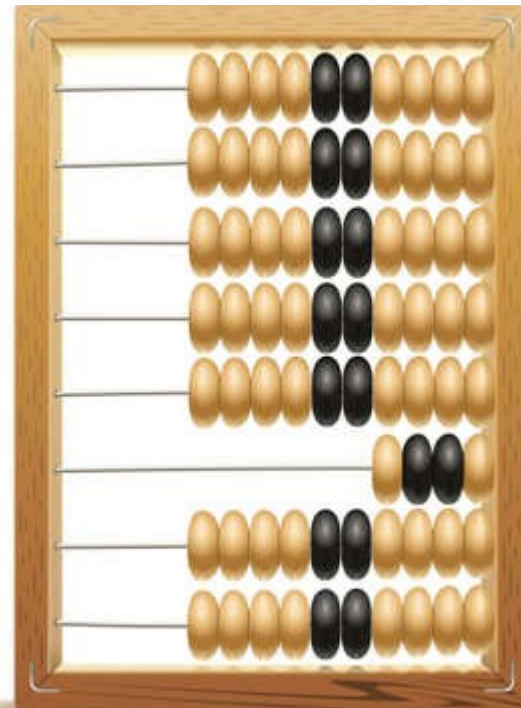


- Combinano uno o più operandi e permettono di valutare “operazioni”
 - Aritmetiche
 - Relazionali
 - Logiche
 - Bit a bit
 - Assegnamento
 - Condizionali e speciali
- A seconda del numero di operandi coinvolti:
 - Unari
 - Binari
 - Ternari

- Piccolo preambolo
 - Statement \rightarrow “Fai qualcosa” ovvero azione
 - Expression \rightarrow “Valuta qualcosa” ovvero “calcolo”
- Le espressioni sono combinazioni di
 - Operatori
 - Operandi

$$A = B * C + D * 17 / K$$


- Tutti binari:
 - $+$ \rightarrow somma
 - $-$ \rightarrow sottrazione
 - $*$ \rightarrow moltiplicazione
 - $/$ \rightarrow divisione
 - $\%$ \rightarrow resto della divisione intera (modulo)
 - $++$ \rightarrow incremento di 1
 - $--$ \rightarrow decremento di 1



- Operatori di incremento o decremento
 - Incremento variabile “a” di 1 $\rightarrow ++a$;
 - Decremento variabile “a” di 1 $\rightarrow --a$;
- Operazioni comuni
- **Caveat:** comportamento differente se prefisso o postfisso
 - $++a \neq a++$
 - Incrementano entrambi “a” ma se in espressioni:
 - Prefisso \rightarrow eseguo incremento prima di altre operazioni
 - Suffisso \rightarrow eseguo incremento dopo tutte le altre operazioni

- Operazioni che hanno come risultato:
 - Vero $\rightarrow 1$ (diverso da \emptyset)
 - Falso $\rightarrow \emptyset$
- $== \rightarrow$ controllo uguaglianza (no virgola mobile!)
- $!= \rightarrow$ controllo differenza (non esiste \neq)
- $>$
- $<$
- $>=$
- $<=$

- Operatori il cui risultato è vero(1)/falso(0)
- `&&` → and logico
- `||` → or logico
- `!` → not logico

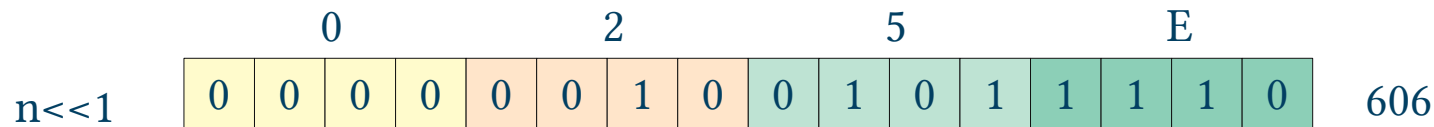
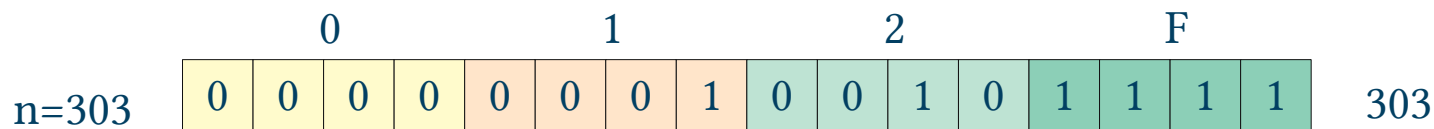
&&	!alive	alive
!dead		
dead		

- Permettono assegnamento valore a variabile
 - $=$ \rightarrow assegnamento
 - $+=$ \rightarrow assegnamento con somma
 - $-=$ \rightarrow ...
 - $*=$
 - $\%=$
 - ...

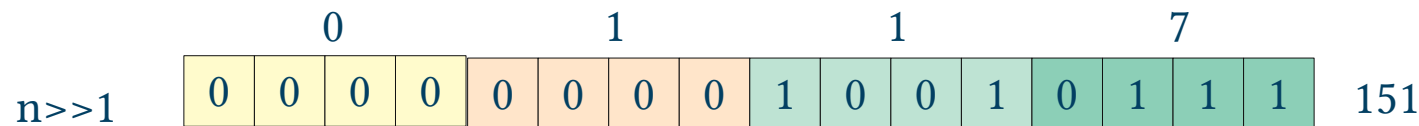
- Condizionali
 - ?
 - Unico operatore ternario del C
- Speciali
 - Li dettaglieremo man mano che li incontreremo
 - sizeof() dimensione di un dato
 - & indirizzo di una variabile
 - * puntatori o accesso memoria
 - , concatenazione di espressioni
 - [] indice array
 - ...

- Lavorano a livello dei singoli bit di ciascun dato
 - $\&$ \rightarrow and bit a bit
 - $|$ \rightarrow or bit a bit
 - \sim \rightarrow not bit a bit
 - \wedge \rightarrow xor bit a bit
 - \ll \rightarrow left shift
 - \gg \rightarrow right shift

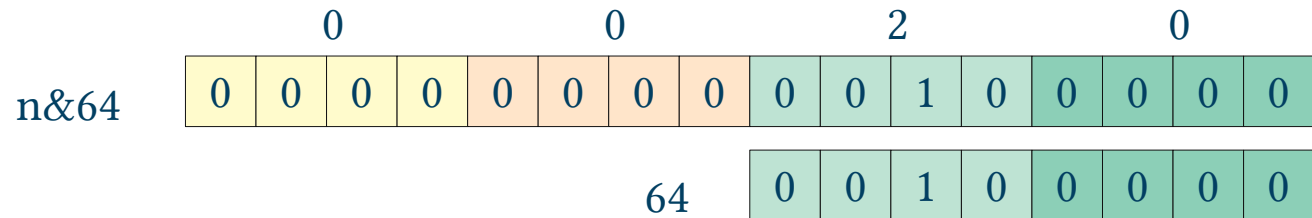
Operatori bit a bit



Da destra entrano zeri → “multiplico per 2 ad ogni bit di spostamento”



Da sinistra replico bit più significativo (shift aritmetico)
Divido per 2 ad ogni spostamento?



- Combinazioni di operatori e operandi
- Esempi:
 - 17 → espressione senza operandi
 - $17 * 2 + 3$ → espressione costante
 - $a = b * 17 + c / 4 + e \% 5 * 2$ → calcolata in esecuzione

- Operatori differenti possono avere precedenze differenti
 - $3*4+2$
 - Nella valutazione delle espressioni si parte dagli operatori con maggiore precedenza
- A parità di precedenza gli operatori vengono valutati in base alla associatività
 - $a\%b\%c$ (\rightarrow)
 - $a = b = c;$ (\leftarrow)
- **Nel dubbio ()**

Operator	Description	Associativity
() [] . -> ++ --	Parentheses or function call Brackets or array subscript Dot or Member selection operator Arrow operator Postfix increment/decrement	left to right
++ -- + - ! ~ (type) * & sizeof	Prefix increment/decrement Unary plus and minus not operator and bitwise complement type cast Indirection or dereference operator Address of operator Determine size in bytes	right to left
* / %	Multiplication, division and modulus	left to right
+ -	Addition and subtraction	left to right
<< >>	Bitwise left shift and right shift	left to right
< <= > >=	relational less than/less than equal to relational greater than/greater than or equal to	left to right
== !=	Relational equal to or not equal to	left to right
&&	Bitwise AND	left to right
^	Bitwise exclusive OR	left to right
	Bitwise inclusive OR	left to right
&&	Logical AND	left to right
	Logical OR	left to right
? :	Ternary operator	right to left
= += -= *= /= %= &= ^= = <<= >>=	Assignment operator Addition/subtraction assignment Multiplication/division assignment Modulus and bitwise assignment Bitwise exclusive/inclusive OR assignment	right to left
,	comma operator	left to right



UNIVERSITÀ DI PARMA

Le Espressioni



KEEP
CALM
IT'S
QUESTION
TIME

We must either institute conventional forms of expression or else pretend that we have nothing to express.

George Santayana, Soliloquies in England