

# “Conta” ispirata a una narrazione di Giuseppe Flavio

Claudio Mirolo

Dipartimento di Scienze Matematiche, Informatiche e Fisiche,  
Università di Udine, via delle Scienze 206 – Udine

[claudio.mirolo@uniud.it](mailto:claudio.mirolo@uniud.it)

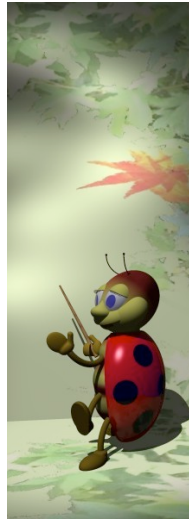
Laboratorio di Programmazione

[nid.dimi.uniud.it](http://nid.dimi.uniud.it)



# Sommario

- 1 programmazione object-oriented
- 2 modelli
- 3 epilogo



# Un miracolo. . .

Il primo miracolo è che combinare un gran numero di volte un piccolo numero di operazioni elementari permette una potenza d'azione considerevole.

*Jacques Mazoyer, 2005*



# Un miracolo. . .

Il primo miracolo è che  
combinare un gran numero di volte  
un piccolo numero di operazioni elementari  
permette una potenza d'azione considerevole.

*Jacques Mazoyer, 2005*



# Un miracolo. . .

Il primo miracolo è che combinare un gran numero di volte un piccolo numero di operazioni elementari permette una potenza d'azione considerevole.

*Jacques Mazoyer, 2005*



# Epistemologia procedurale

La rivoluzione informatica  
è una rivoluzione nel modo di pensare  
e di esprimere quello che si pensa.  
L'essenza di questo cambiamento è  
l'emergere [di un'] *epistemologia procedurale*.

[...] i programmi devono essere scritti  
affinché li possano leggere le persone,  
e solo incidentalmente  
per farli eseguire dalle macchine.

*Gerald J. Sussman, 2004*



# Epistemologia procedurale

La rivoluzione informatica  
è una rivoluzione nel modo di pensare  
e di esprimere quello che si pensa.  
L'essenza di questo cambiamento è  
l'emergere [di un'] *epistemologia procedurale*.

[...] i programmi devono essere scritti  
affinché li possano leggere le persone,  
e solo incidentalmente  
per farli eseguire dalle macchine.

*Gerald J. Sussman, 2004*



# Epistemologia procedurale

La rivoluzione informatica  
è una rivoluzione nel modo di pensare  
e di esprimere quello che si pensa.  
L'essenza di questo cambiamento è  
l'emergere [di un'] *epistemologia procedurale*.

[...] i programmi devono essere scritti  
affinché li possano leggere le persone,  
e solo incidentalmente  
per farli eseguire dalle macchine.

*Gerald J. Sussman, 2004*





# Epistemologia procedurale

La rivoluzione informatica  
è una rivoluzione nel modo di pensare  
e di esprimere quello che si pensa.  
L'essenza di questo cambiamento è  
l'emergere [di un'] *epistemologia procedurale*.

[...] i programmi devono essere scritti  
affinché li possano leggere le persone,  
e solo incidentalmente  
per farli eseguire dalle macchine.

*Gerald J. Sussman, 2004*



# Programmazione

Competenze nella programmazione

=

Capacità di *organizzazione* (mentale)

e di *chiarezza* nell'esprimere l' *organizzazione*



# Programmazione

Competenze nella programmazione

=

Capacità di *organizzazione* (mentale)

e di *chiarezza* nell'esprimere l'organizzazione



# Programmazione

Competenze nella programmazione

=

Capacità di *organizzazione* (mentale)

e di *chiarezza* nell'esprimere l' *organizzazione*



# Programmazione

Competenze nella programmazione

=

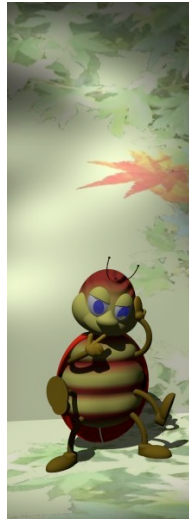
Capacità di *organizzazione* (mentale)

e di *chiarezza* nell'esprimere l' *organizzazione*



# Sommario

- 1 programmazione object-oriented
- 2 modelli
- 3 epilogo



# Linguaggi di programmazione imperativa

1945–55		Linguaggi macchina, linguaggi assembleri
1953	<b>FORTRAN</b>	J.W. Backus: progetto del primo linguaggio di programmazione ad alto livello
1957		Primo compilatore FORTRAN completo
1958–68	<b>Algol</b>	F.L. Bauer, J.W. Backus, P. Naur, E.W. Dijkstra, C.A.R. Hoare e molti altri: eleganza e pulizia <i>“Un linguaggio così avanzato per il suo tempo da migliorare non solo quelli precedenti, ma anche quasi tutti i successivi.”</i> (Hoare, 1973)
1959	<b>COBOL</b>	Linguaggio orientato al software gestionale
1969	<b>Pascal</b>	N. Wirth: linguaggio “Algol-like”
1969–73	<b>C</b>	D. Ritchie: legato al sistema operativo Unix



# Linguaggi di programmazione funzionale

1958	LISP	J. McCarthy (progetto), S. Russel, T. Hart & M. Levin (realizzazione) — fra i linguaggi ad alto livello è preceduto solo dal FORTRAN
1962		Primo compilatore LISP completo
anni '70	Schemer	G.L. Steele Jr. & G.J. Sussman ...
	Scheme	Sviluppi di LISP: progetto sperimentale di un linguaggio di nuova concezione
anni '70	ML	R. Milner: linguaggio funzionale con un sofisticato sistema di tipi (polimorfi)
1990	Haskell	Puramente funzionale con tipi polimorfi





# Linguaggi di programmazione orientata agli oggetti

anni '60	Simula	O.-J. Dahl & K. Nygaard: problemi di simulazione — introduzione dei principali concetti
anni '70	Smalltalk	A. Kay: intuizione della generalità ed espressività dell'approccio OOP
anni '80		estensioni OO di vari linguaggi (C, Pascal, LISP, ...)
1985	C++	B. Stroustrup: linguaggio di ampia diffusione
1988	Eiffel	B. Meyer: formalizzazione nel linguaggio degli "invarianti di classe"
1991	Oak	J. Gosling: Java in embrione
1995	Java	Multipiattaforma, rete, sicurezza. . .



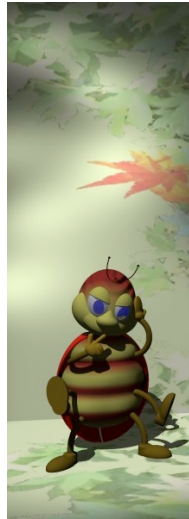
# Linguaggi di programmazione logica

anni '60		Precursori “question-answering” ispirati da “advice taker”, ipotizzato da J. McCarthy (1958)
1960–70		Sviluppi nell’ambito dell’intelligenza artificiale
1969	Absys	J.M. Foster & E.W. Elcock: primo linguaggio asserzionale (dichiarativo)
1972	Prolog	R. Kowalski (modello di calcolo: risoluzione SL), A. Colmerauer & P. Russel (realizzazione): programmazione logica “general-purpose”
anni '90	Curry	M. Hanus e altri: linguaggio logico-funzionale

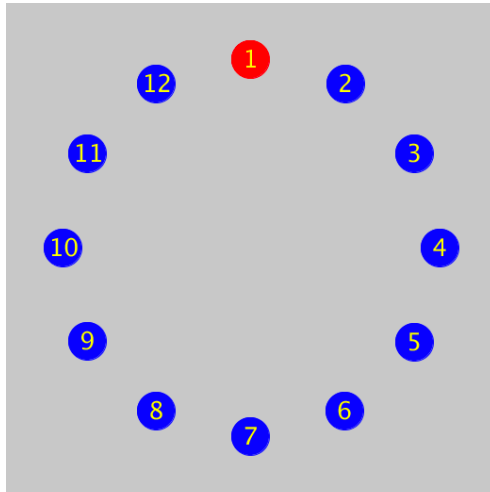


# Sommario

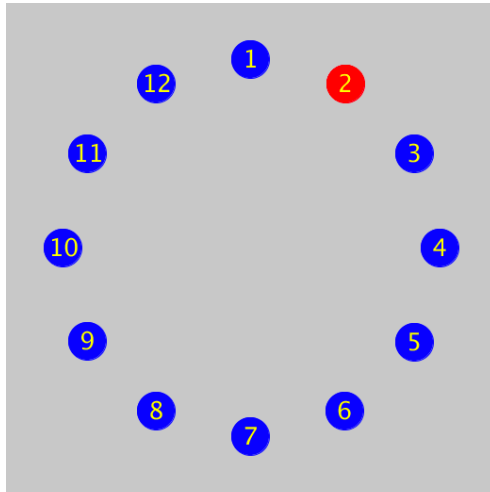
- 1 programmazione object-oriented
- 2 modelli**
- 3 epilogo



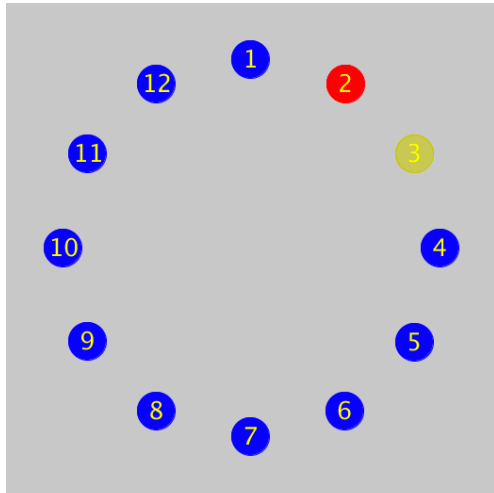
# “Conta”



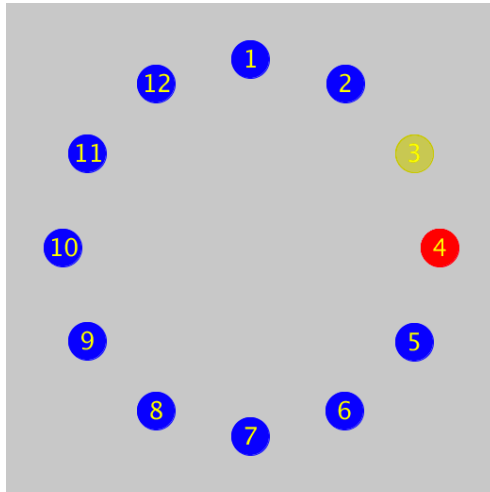
# “Conta”



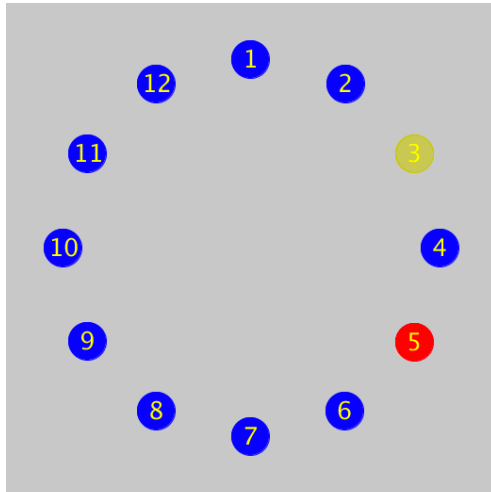
# “Conta”



# “Conta”

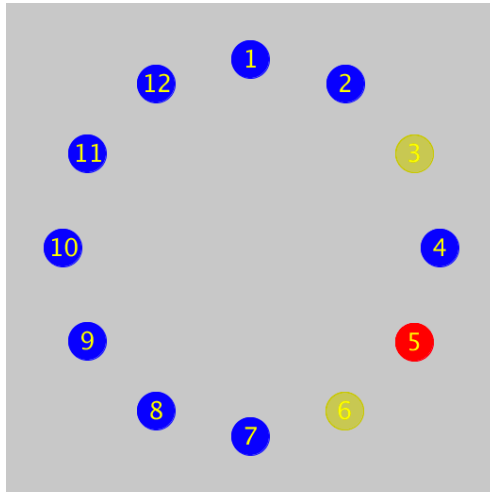


# “Conta”

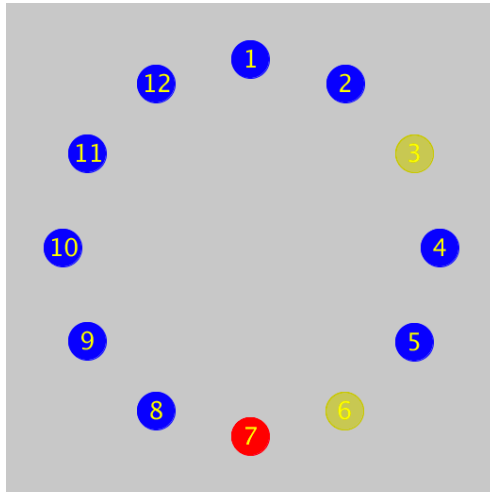




# “Conta”



# “Conta”



# Modelli: Commensale

**Commensale**

numero

identifica

vicino

ricevi

passa

offri

accetta



# Modelli: Commensale

**Commensale**

numero

identifica

vicino

ricevi

passa

offri

accetta



# Modelli: Commensale

**Commensale**

numero

identifica

vicino

ricevi

passa

offri

accetta



# Modelli: Commensale

**Commensale**

numero

identifica

vicino

ricevi

passa

offri

accetta



# Modelli: Commensale

**Commensale**

numero

identifica

vicino

ricevi

passa

offri

accetta



# Modelli: Commensale

**Commensale**

numero

identifica

vicino

ricevi

passa

offri

accetta





# Modelli: Tavolata

## Tavolata

convivialità



# Modelli: Tavolata

**Tavolata**

convivialità



# Al lavoro!

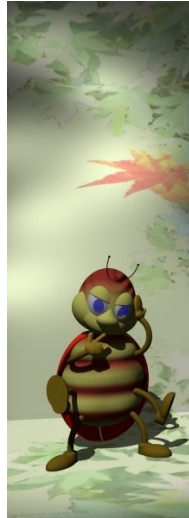


# Al lavoro!



# Sommario

- 1 programmazione object-oriented
- 2 modelli
- 3 epilogo



# Epilogo...

## Giuseppe Flavio: *“La guerra Giudaica”* Libro III - Capitolo 8 (ca. 75 d.C.)

Comunque, malgrado l'estrema tensione, non gli mancò la consueta sagacia; ma confidando nella provvidenza di Dio, mise in gioco la sua vita affidandosi alla sorte:

“E adesso”, disse, “visto che ormai siete decisi a morire, venite e affidiamo al caso [l'ordine] secondo cui ci procureremo mutuamente la morte.” ...



# Epilogo...

Giuseppe Flavio: *“La guerra Giudaica”*  
Libro III - Capitolo 8 (ca. 75 d.C.)

Comunque, malgrado l'estrema tensione, non gli mancò la consueta sagacia; ma confidando nella provvidenza di Dio, mise in gioco la sua vita affidandosi alla sorte:

“E adesso”, disse, “visto che ormai siete decisi a morire, venite e affidiamo al caso [l'ordine] secondo cui ci procureremo mutuamente la morte.” ...



# Epilogo...

*“Colui che sarà sorteggiato per primo, sarà ucciso dal secondo sorteggiato, e così via [...], [in modo che] nessuno di noi perisca per azione della propria mano destra, [e] perché sarebbe ingiusto che, quando gli altri se ne siano già andati, qualcuno si penta e salvi sé stesso.”*

Questa proposta sembrò loro molto giusta; e quando li ebbe convinti a regolare la questione attraverso sorteggi, estrasse un numero anche per sé.





# Epilogo...

*“Colui che sarà sorteggiato per primo, sarà ucciso dal secondo sorteggiato, e così via [...], [in modo che] nessuno di noi perisca per azione della propria mano destra, [e] perché sarebbe ingiusto che, quando gli altri se ne siano già andati, qualcuno si penta e salvi sé stesso.”*

Questa proposta sembrò loro molto giusta; e quando li ebbe convinti a regolare la questione attraverso sorteggi, estrasse un numero anche per sé.



# Epilogo...

[...] Se anche Giuseppe fosse morto con loro, la morte sarebbe sembrata più dolce della vita; ma, che ciò sia dovuto al caso o alla provvidenza di Dio, [Giuseppe] restò per ultimo assieme a un altro.

E poiché voleva assolutamente evitare sia di essere condannato dalla sorte, sia, se fosse rimasto per ultimo, di bagnare la propria mano destra nel sangue di un concittadino, lo persuase a concedergli fiducia e a far sì che entrambi rimanessero vivi.



# Nella rielaborazione di Donald Knuth!



# Nella rielaborazione di Donald Knuth!



# Riferimenti



J. Mazoyer (2005)

L'enseignement de l'informatique ...

*Académie des sciences*



G.J. Sussman (2004)

The Legacy of Computer Science

*Computer Science: Reflections on the Field*



R.L. Graham, D.E. Knuth & O. Patashnik (1988)

Concrete Mathematics

*Addison-Wesley*

