

SSIS 2007

Maria Rita Laganà, Dipartimento di Informatica
Università di Pisa

Incontro I:

Ambienti di programmazione multi-
agente
la ricorsione e il parallelismo

Presentazione

- Pagina web del corso:
www.di.unipi.it/~lagana/ssis/ssis_2007.htm
- Scopo:
Scrivere semplici modelli su computer per esplorare i fenomeni naturali.

Questi esempi illustrano il fenomeno dell'*emergenza* di complessità in semplici sistemi auto-organizzati.

Presentazione di questo modulo

- Piattaforma di programmazione: NetLogo/StarLogo
 - Non viene richiesta nessuna esperienza precedente di programmazione.
 - ... ma un po' di tempo per apprenderla !
- Quando ci incontreremo
 - 6 sabati di cui 3 in laboratorio
- Letture interessanti:
 - Turtles, Termites, and Traffic Jams: Explorations in Massively Parallel Microworlds. Michel Resnick, MIT Press (1994)
 - How Nature Works. Per Bak, Copernicus Books (1999)
- Laboratorio
 - Scelta di un progetto da sviluppare.

Cosa vedremo

- Programmazione in StarLogo
Strumento per modellare sistemi complessi
- Studieremo la teoria dei sistemi complessi?
No.
Ne costruiremo piuttosto qualcuno simulandolo al computer.
- Quale sarà la nostra ricetta per simularli?

Modellare è anche arte e, come tale, richiede pratica e tempo per pensare ...Non è facile dire quando un modello è bello e significativo!!!

Gli attori

- Osservatore

- Abita allo (0, 0)
- Crea tartarughe, traccia i grafici, ...
- Si rivolge alle tartarughe e ai patch

tramite l'istruzione ask

```
ask-turtles [set color yellow]
ask-turtle 0 [fd 1]
ask-patches [setpc blue]
```

- Collezione dati per tartarughe e patch

```
show count-turtles-with [color = blue]
show count-patches-with [pc = yellow]
set totale sum-of-turtles [energia]
```

Gli attori

- Tartarughe
 - Variabili: who, xcor, ycor, heading, color, shape, etc.
 - Azioni: fd, jump, stamp, hatch, die
 - Ma possiamo aggiungere variabili:
turtles-own [eta]
- Patch
 - Variabili: xcor, ycor, color, etc.
 - Azioni: sprout
 - Ma possiamo aggiungere variabili
patches-own [profumo]

Gli attori

Agenti

```
turtles, turtle 2,  
breeds  
patches patch-at 1 0  
turtles-with [pc = blue]  
turtles-with [xcor > 0 and ycor > 0]
```

Comandi: *Observer*, *Turtle*, e *Patch*.

provare show 2 + 2 come osservatore e come tartarughe.

Random

Prova:

```
random 10  
random 10  
random 10
```

Prova:

```
set-random-seed 134  
random 10  
random 10  
set-random-seed 124  
random 10  
random 10
```


StarLogo

- **ca** cancella tutto
- **crt 10** crea 10 tartarughe
- **fd 10** muove le tarta di 10 passi
- **seth 90** inclina la testa delle tarta a 90 gradi (guardano a est)
- **rt 45** gira la testa delle tarta di 45 gradi a destra
- **stamp red** il patch sotto le tarta diventa rosso
- **setcolor green** colora le tarta di verde
- **setxcor 0** le tarta sono sull'asse y.
- **setycor screen-height** posiziona le tarta sull'asse x
- **setycor screen-height - ycor** simmetria...
- **setxy mouse-xcor mouse-ycor** posiziona le tarta sul mouse
- **seth (random 360)** la testa delle tarta è piegata casualmente
- **pd** mette giù la penna (**pu** la solleva)
- **die** le tarta muiono

In rosso i comandi sulla finestra osservatore, in blu quelli sulla finestra tartarughe

StarLogo: ask-patches []

- `setpc red` colora di rosso i patch
- `setpc pc + 1` il colore è un numero....
- `setpc random 256`
- `setpc xcor` i patch con la stessa x ...
- `setpc xcor + ycor` i patch sulle diagonali ...
- `repeat 100 [setpc pc + 1]`
- `if xcor < 0 [setpc red]` colora in base alla x
- `if (distance 5 5) < 10 [setpc green]` un cerchio.
- `if (distance mouse-xcor mouse-ycor) < 4 [setpc blue]`
- `ifelse ycor < 0 [setpc red] [setpc blue]`

In rosso i comandi sulla finestra osservatore, in blu quelli sulla finestra tartarughe

StarLogo: interfaccia

- **Bottoni**
 - Per eseguire un comando
 - Per eseguire un comando per sempre
- **Slider**
 - Per assegnare valori a variabili
- **Monitor**
 - Per controllare i valori assunti di una variabile
- **Paint**
 - Per colorare la griglia e le tartarughe
- ...

Variabili StarLogo

- **Globali** (valori assegnabili con slider):
globals [clock number]
- **Turtle:** turtles-own [energia]
- **Patch:** patches-own [chimica]
- **Local Variables** (nelle procedure)
locals [temp1 temp2]

Nota Le variabili patch sono condivise dalle tartarughe e dunque una variabile turtle e una variabile patch non possono avere lo stesso nome.

Un semplice programma

```
to inizia
ca
crt 100
ask-turtles [setshape turtle-shape
set xcor random screen-width
set ycor random screen-height ]
end
```

```
to passeggia
ask-turtles [
rt random 50
lt random 50
fd 1]
end
```

La 1-geometria NetLogo

Sistemare lo schermo e la dimensione delle tartarughe: `set size 30`

Diciamo **stato** della tartaruga la sua posizione e la sua inclinazione.

Osservazione

- ❖ Se un cammino si chiude in modo trasparente la tartaruga compie una rotazione totale multipla di 360. Viceversa? Riflettere sul caso:

```
repeat n [fd passo rt angolo]
```

Quesito

- ❖ Perché siamo sempre su una circonferenza?

La 1-geometria NetLogo

repeat n [fd passo rt angolo]

- ❖ Se conosciamo α possiamo predire il numero di lati? In base alla osservazione precedente occorre sapere di quanto moltiplicare α per avere il primo multiplo di 360.
- ❖ per i fattori di 360 che non collaborano al $\text{MCD}(\alpha, 360)$

La 1-geometria NetLogo

Se $\alpha = 144$, $\text{MCD}(\alpha, 360) = 72$. Il poligono è a 5 lati.

Se $\alpha = 135$, $\text{MCD}(\alpha, 360) = 45$. Il poligono è a 8 lati.

Se $\alpha = 108$, $\text{MCD}(\alpha, 360) = 36$. Il poligono è a 10 lati.

Se $\alpha = 72$, $\text{MCD}(\alpha, 360) = 36$. Il poligono è a 5 lati.

Se $\alpha = 35$, $\text{MCD}(\alpha, 360) = 5$. Il poligono è a 72 lati.

Quesiti

❖ Che differenza c'è tra $360/N$ e $720/N$?

$$N * \alpha = 360 * R$$

ha anche soluzioni per cui N e R non sono primi fra loro

La 1-geometria NetLogo

- ❖ La rotazione totale lungo un cammino chiuso **semplice** (senza incroci) è +360 o -360.
 - ❖ Teorema difficile dimostrazione

repeat n [fd passo rt angolo]

- ❖ Come facciamo a indicare un poligono semplice di N lati?
 - ❖ Basta che $\alpha = \text{angolo} = 360/N$
- ❖ **Teorema degli angoli esterni**
 - ❖ La somma degli angoli "esterni" di un poligono semplice è 360.

Una stella di pentagoni

- `repeat 5 [fd 40 rt 72]` costruisce un pentagono.
- Quando si chiude il percorso
`rt 35 repeat 5 [fd 40 rt 72]`?
- Usare il programma `pentagoni`.
- Si noti che $\text{mcm}(35, 360) = 2520$; $\text{MCD} = 5$ e
 $2520/35 = 360/5 = 72$
- Usare il programma `pentastar`

pentagoni in parallelo

- Basta creare il numero opportuno di tartarughe e disporle adeguatamente
- Usare il programma pentastar

```
to inizia
```

```
ca crt 72
```

```
ask-turtles[vai ]
```

```
end
```

```
to vai
```

```
seth who * 35
```

```
setc who
```

```
pd
```

```
pentagono
```

```
end
```

```
to pentagono
```

```
repeat 5 [fd 100 rt 72]
```

```
end
```

Esempi ricorsivi

- Una definizione si dice ricorsiva se si basa sul Principio d'induzione matematica
 - Si definisce un caso base in funzione di un parametro n ($n = n_0$)
 - Si definisce il caso generico n in funzione di $n-1$
 - \rightarrow la definizione vale per ogni $n > n_0$
- Esempi
 - Il fattoriale
 - $n! = 1$ se $n = 0$
 - $n! = n * (n - 1)!$ se $n > 0$
 - La successione di Fibonacci
 - $F(0) = 1$
 - $F(1) = 1$
 - $F(n) = F(n - 1) + F(n - 2)$ se $n > 1$

Colori, colori...

```
to setup  
ca  
crt 10  
ask-turtles [pd]  
end
```

```
to go  
rt 1  
hatch [setc who / 30  
repeat 36 [fd 15 rt 10]  
die]  
end
```

```
repeat 2000 [go]  
repeat 10 [ask-turtles[go]]  
ask-turtles [repeat 10[go]]
```