

Il gioco “Life”: la matematica della vita

di Alessandro Musesti

Università Cattolica del Sacro Cuore, Brescia

30 gennaio 2018



UNIVERSITÀ
CATTOLICA
del Sacro Cuore



CORSO DI LAUREA TRIENNALE IN

MATEMATICA con curricula in **Matematica, Fisica e Informatica** (CLASSE L35)

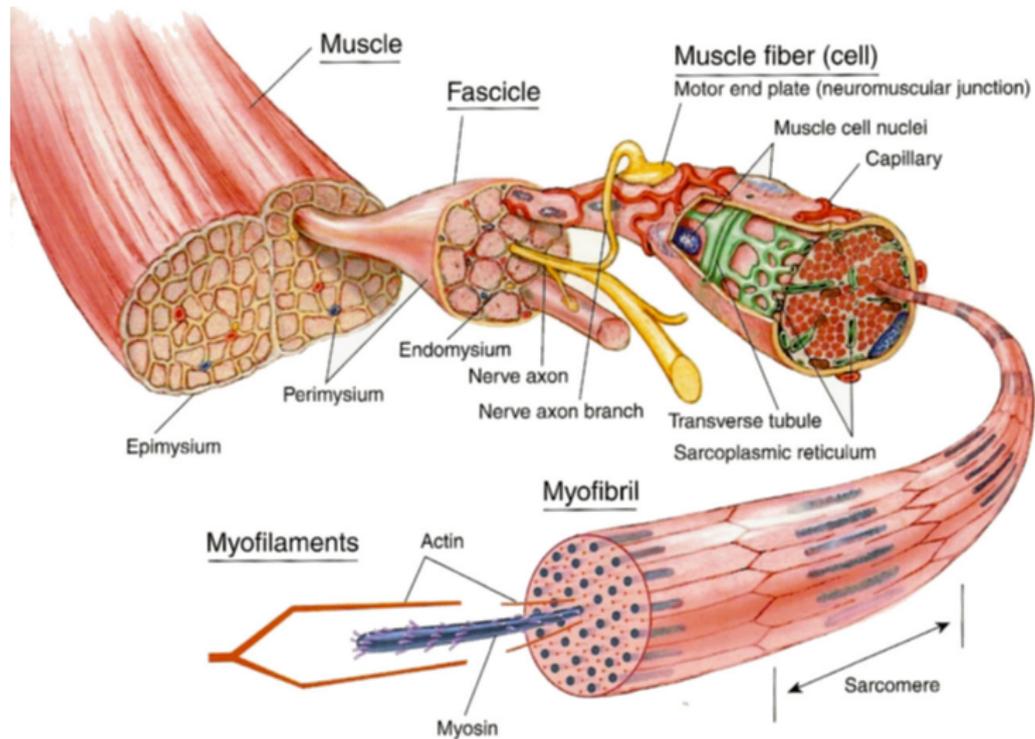
Il corso di laurea triennale realizza tre distinti percorsi formativi:

- un curriculum in **Matematica**
- un curriculum in **Fisica**
- un curriculum in **Informatica**, di nuova istituzione

In particolare, i curricula in Fisica e in Informatica assicurano una **formazione equivalente** rispettivamente a una laurea triennale in Fisica e a una laurea triennale in Scienze dell'informazione.

Tutti i curricula forniscono approfondite e moderne conoscenze di base nell'area della matematica, focalizzando nel contempo la propria attenzione nei rispettivi percorsi sui distinti ambiti disciplinari della matematica, della fisica e dell'informatica.

Skeletal muscle tissue



The passive model

Ehret, Böl, Itskov, JMPS, 59 (2011)

$$W_{pas}(C) = \frac{\mu}{4} \left[\frac{1}{\alpha} (e^{\alpha(I_p-1)} - 1) + K_p - 1 \right]$$

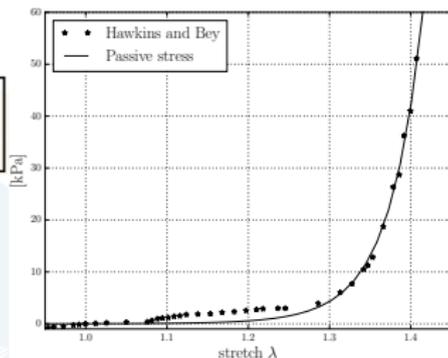
$$I_p = w_0 \frac{\text{tr } C}{3} + (1 - w_0) \text{tr } CM$$

$$K_p = w_0 \frac{\text{tr } C^{-1}}{3} + (1 - w_0) \text{tr } C^{-1}M$$

$$C = F^T F, \quad \det C = 1, \quad M = \mathbf{m} \otimes \mathbf{m}$$

The energy is polyconvex and coercive.

The parameters μ, α, w_0 can be fitted with experimental data on the relation between stress and strain in passive skeletal muscle

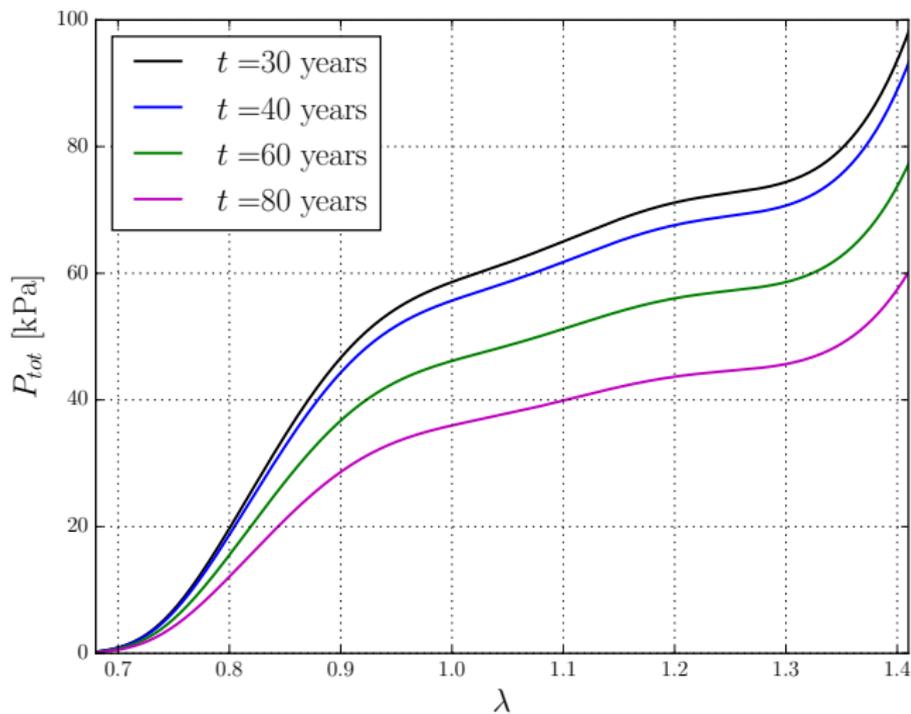


$$\mu = 0.1599 \text{ kPa}$$

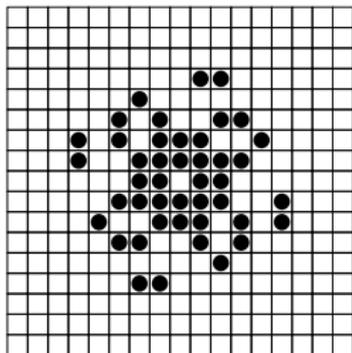
$$\alpha = 19.35$$

$$w_0 = 0.7335$$

Effects of the loss of mass

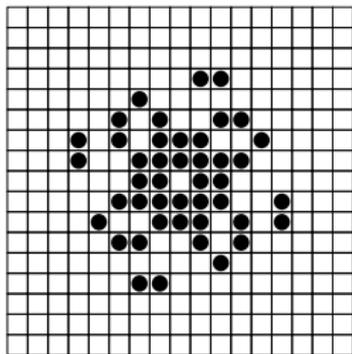


Il gioco 'Life' di Conway



Life fu inventato da Conway alla fine degli anni sessanta del secolo scorso.

Il gioco 'Life' di Conway

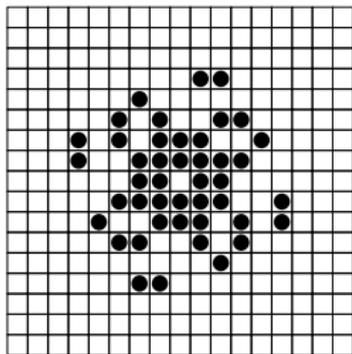


Life fu inventato da Conway alla fine degli anni sessanta del secolo scorso.



John Horton Conway, nato nel 1937, è un matematico inglese, specializzato in teoria dei gruppi (che è una parte dell'Algebra). È uno straordinario inventore di giochi e rompicapi, che di solito studia e risolve piuttosto velocemente.

Il gioco 'Life' di Conway



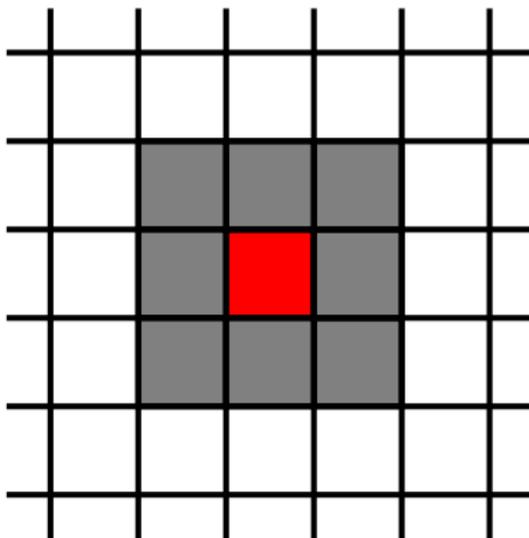
Life fu inventato da Conway alla fine degli anni sessanta del secolo scorso.



John Horton Conway, nato nel 1937, è un matematico inglese, specializzato in teoria dei gruppi (che è una parte dell'Algebra). È uno straordinario inventore di giochi e rompicapi, che di solito studia e risolve piuttosto velocemente. Quando ha inventato Life, però, non credeva certamente che il gioco potesse contenere quel mondo che poi vedremo.

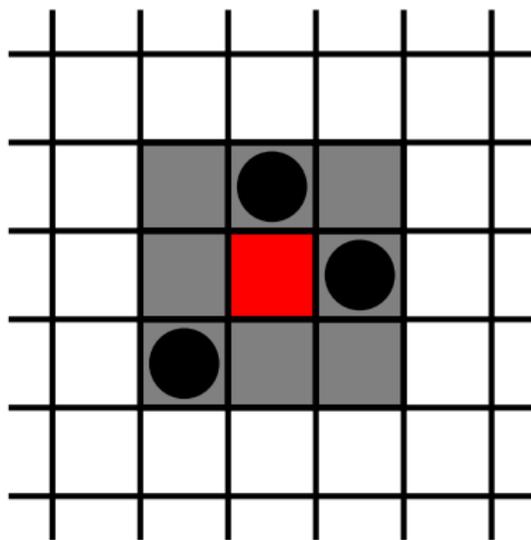
Le regole del gioco

Il “mondo” è una griglia bidimensionale infinita e le “cellule” possono occupare le caselle di questa griglia.



La vita nella casella rossa centrale dipende dalle otto caselle vicine. La regola di Life è denotata da B3/S23.

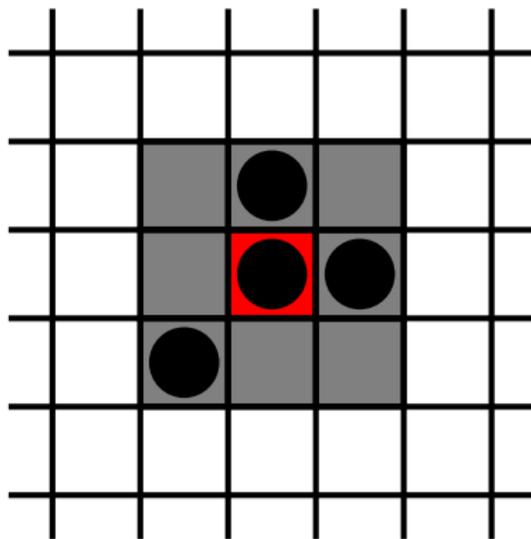
Le regole del gioco



Regola 1: NASCERE

Se una casella vuota è vicina **esattamente** a tre cellule vive, in essa nascerà una cellula (le cellule si riproducono)

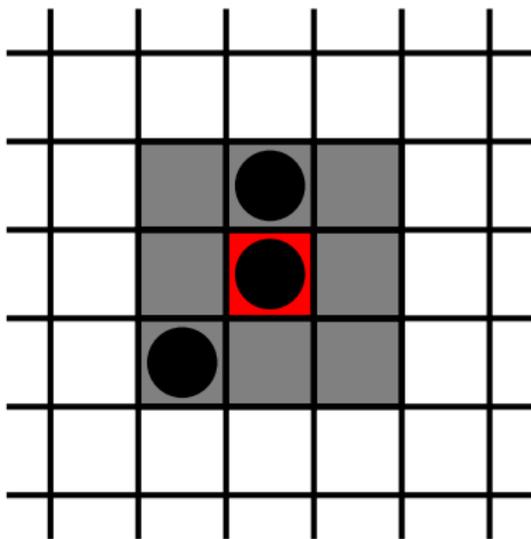
Le regole del gioco



Regola 1: NASCERE

Se una casella vuota è vicina **esattamente** a tre cellule vive, in essa nascerà una cellula (le cellule si riproducono)

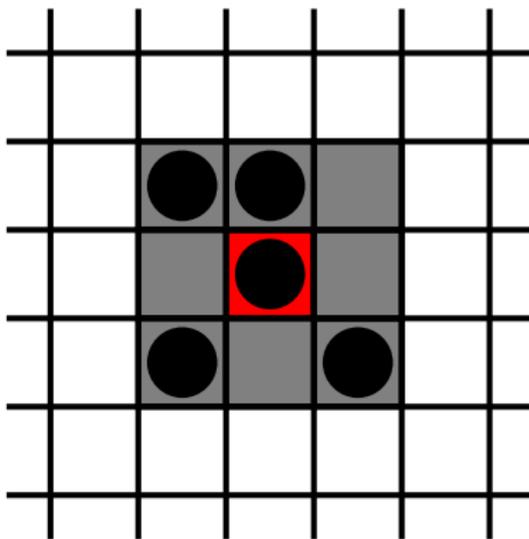
Le regole del gioco



Regola 2: VIVERE

Se una cellula viva è vicina a 2 o 3 cellule vive, essa continua a vivere (le cellule si aiutano a vicenda)

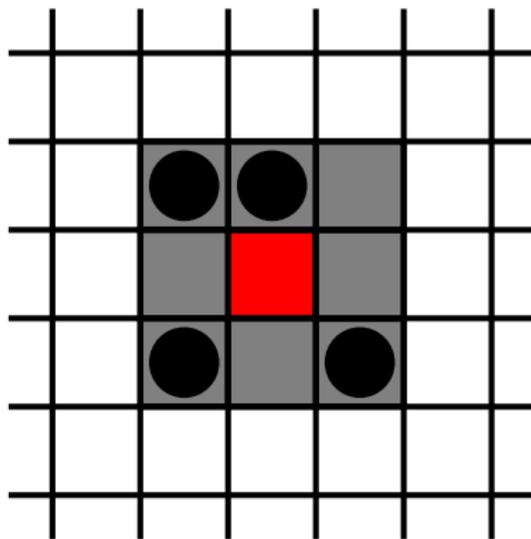
Le regole del gioco



Regola 3: MORIRE

Se una cellula è vicina a 4 o più cellule vive, o a meno di 2 cellule, essa morirà (per sovraffollamento o isolamento)

Le regole del gioco

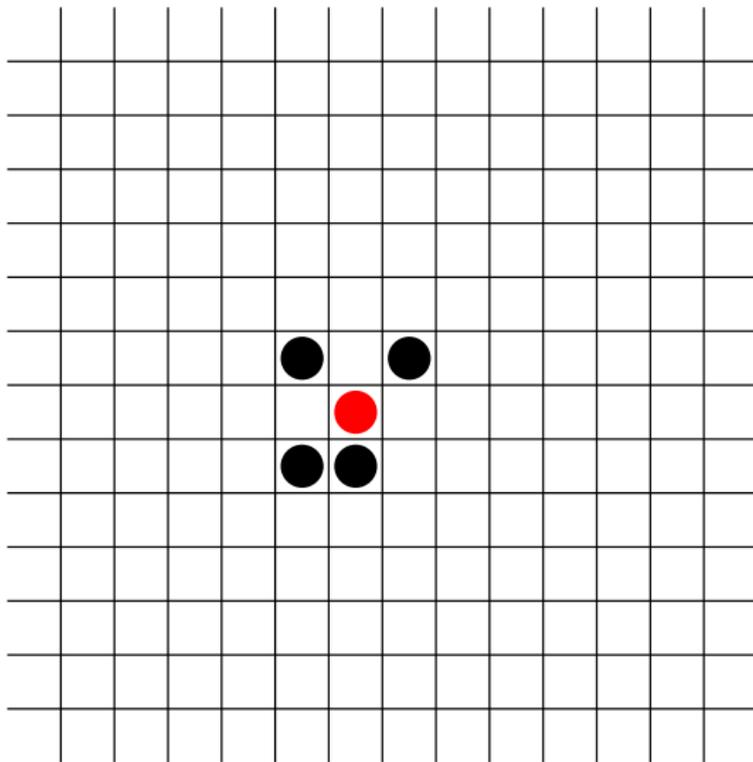


Regola 3: MORIRE

Se una cellula è vicina a 4 o più cellule vive, o a meno di 2 cellule, essa morirà (per sovraffollamento o isolamento)

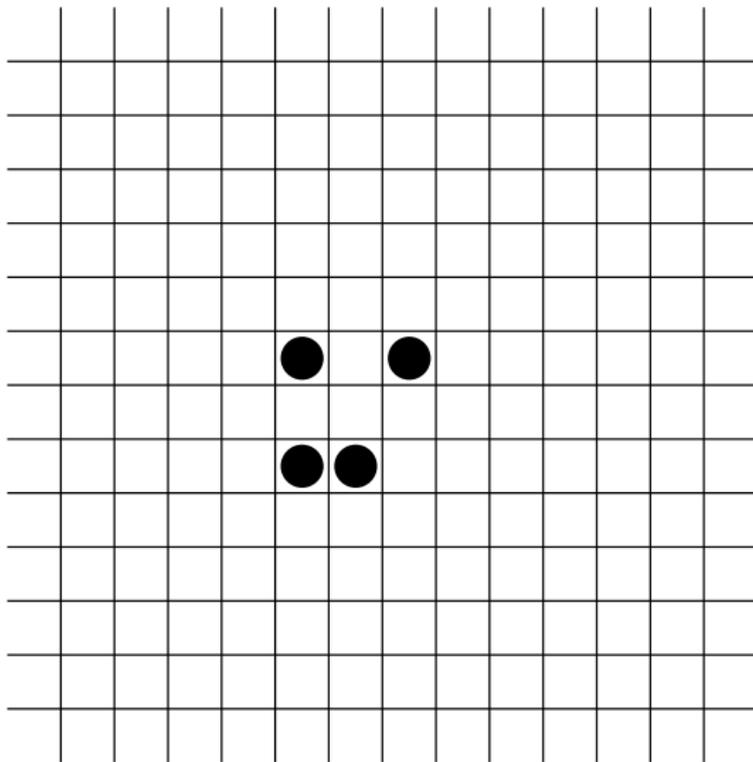
Esempi

La cellula muore



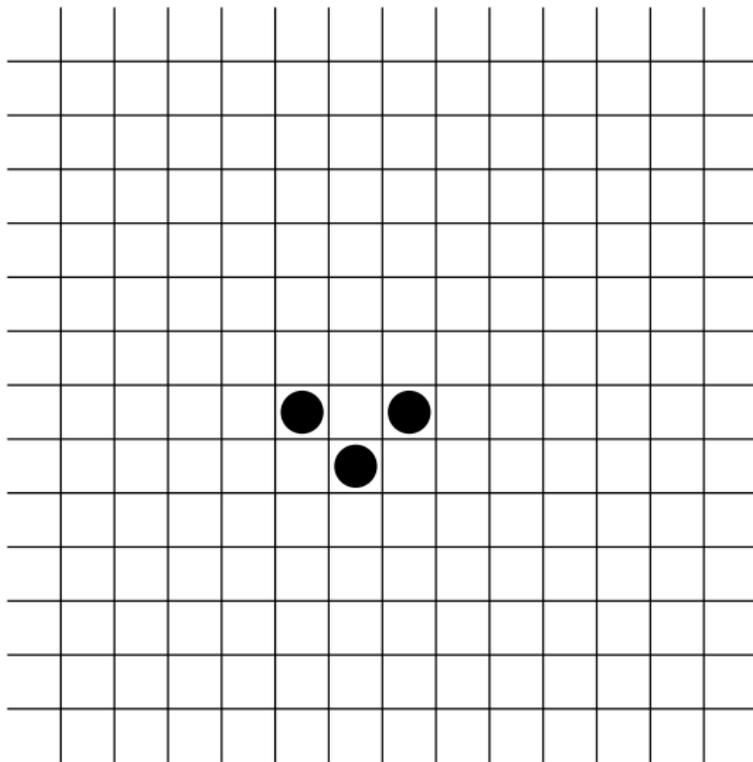
Esempi

La cellula muore



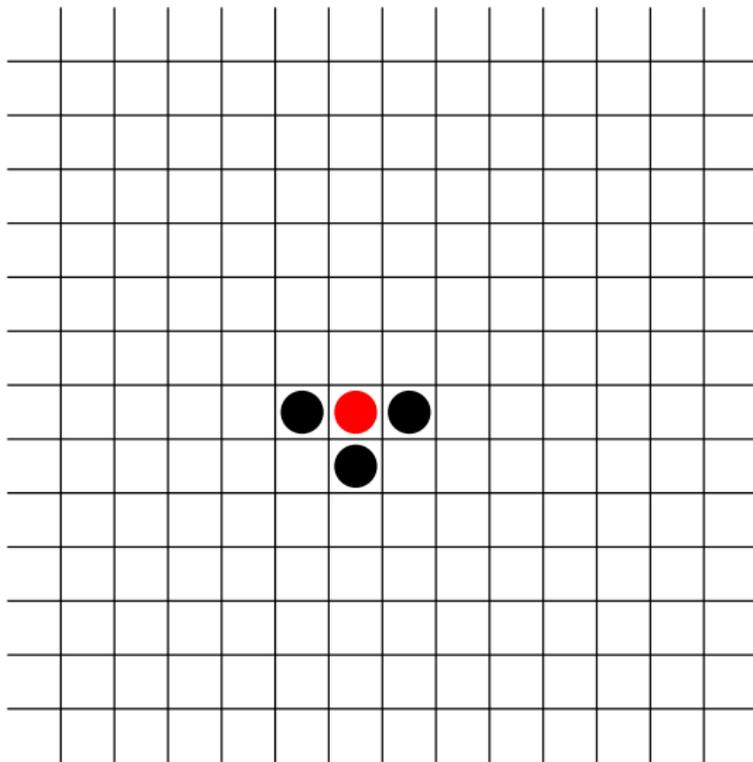
Esempi

La cellula nasce



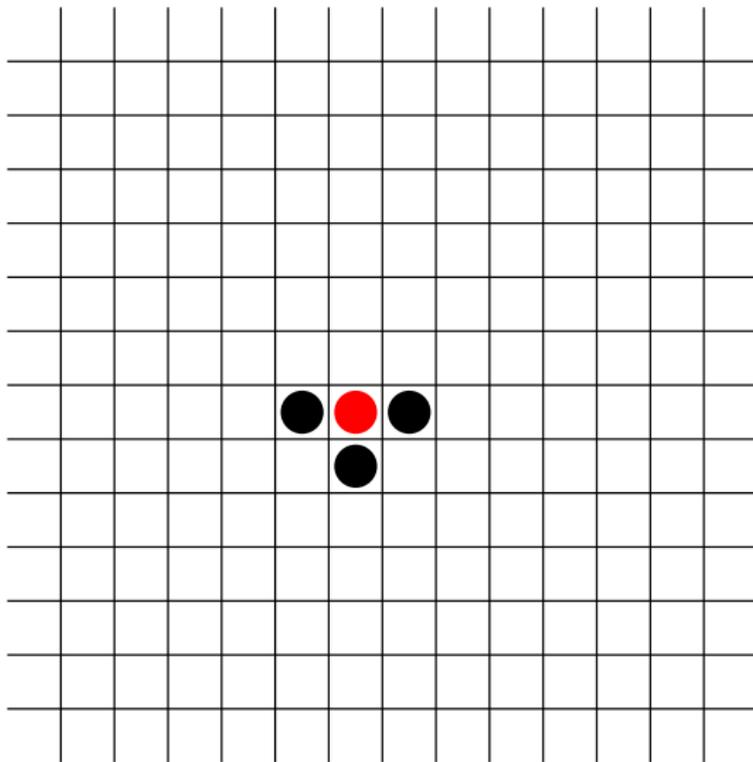
Esempi

La cellula nasce



Esempi

La cellula sopravvive



Life su Scientific American

Ma *Life* è diventato famoso a partire dall'ottobre del 1970, quando Martin Gardner lo descrisse nella sua popolare rubrica "Mathematical Games" della rivista "Scientific American", ponendo anche qualche quesito non ancora risolto.

Life su Scientific American

Ma Life è diventato famoso a partire dall'ottobre del 1970, quando Martin Gardner lo descrisse nella sua popolare rubrica "Mathematical Games" della rivista "Scientific American", ponendo anche qualche quesito non ancora risolto.

Il gioco ebbe tale successo che Martin Gardner ne scrisse in modo più approfondito nel febbraio del 1971.

Life su Scientific American

Ma Life è diventato famoso a partire dall'ottobre del 1970, quando Martin Gardner lo descrisse nella sua popolare rubrica "Mathematical Games" della rivista "Scientific American", ponendo anche qualche quesito non ancora risolto.

Il gioco ebbe tale successo che Martin Gardner ne scrisse in modo più approfondito nel febbraio del 1971.

Gardner scrive:

"My 1970 column on Conway's Life met with such an instant enthusiastic response among computer hackers around the world that their mania for exploring Life forms was estimated to have cost the nation millions of dollars in illicit computer time."

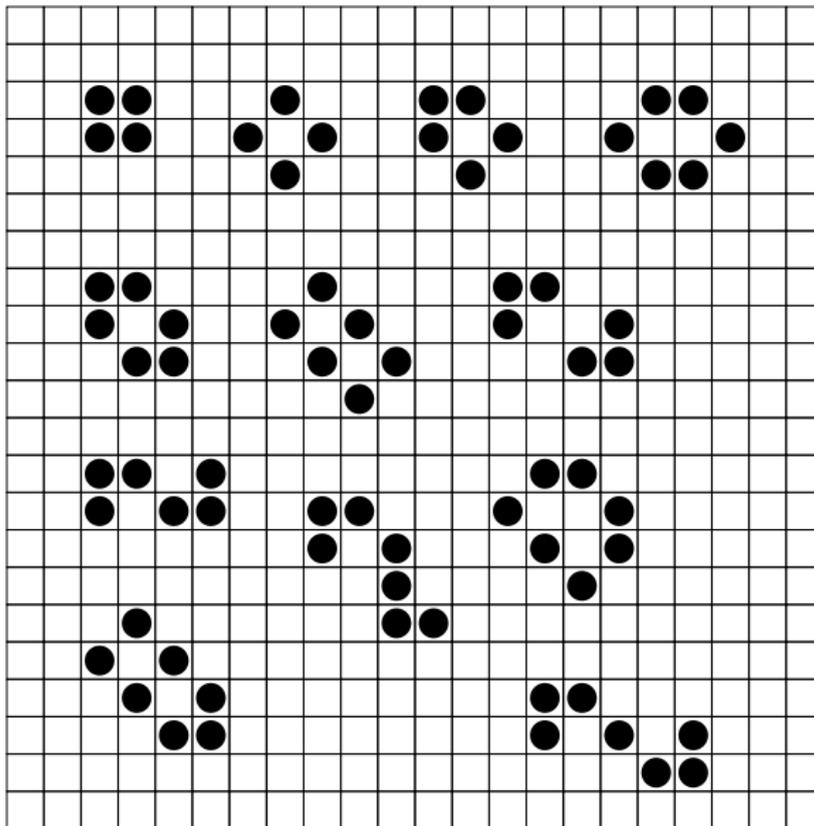
Un po' di classificazione

All'inizio, John Conway e molti altri hanno cercato di classificare i possibili esiti delle configurazioni iniziali di Life.

Tra le più semplici "forme di vita" troviamo le *still life* (nature morte): sono configurazioni stabili, che non cambiano nel tempo. Ce ne sono due con 4 celle, una con 5 celle, cinque con 6 celle e quattro con 7 celle.

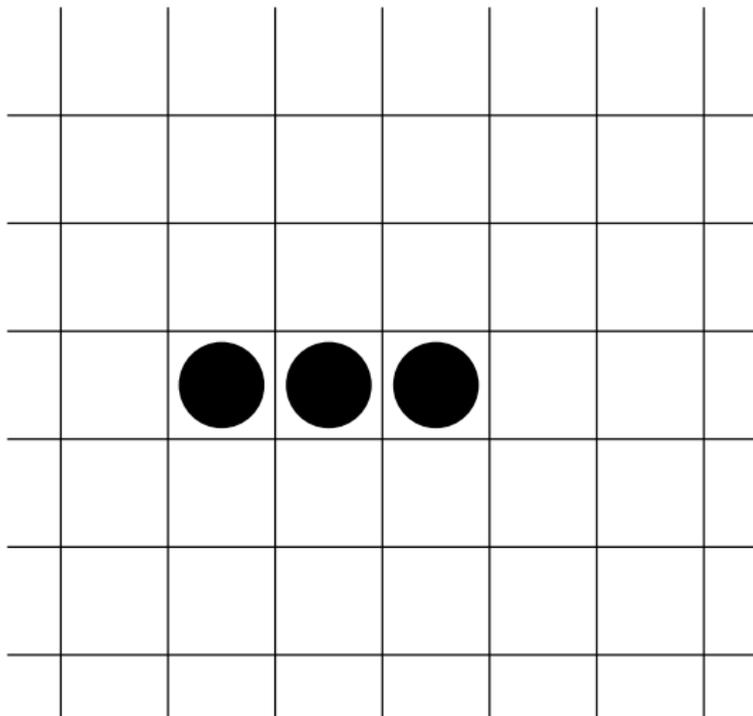
Vengono dati nomi strani a queste semplici "forme di vita": *blocco, vasca da bagno, alveare, barca, nave...*

Still life



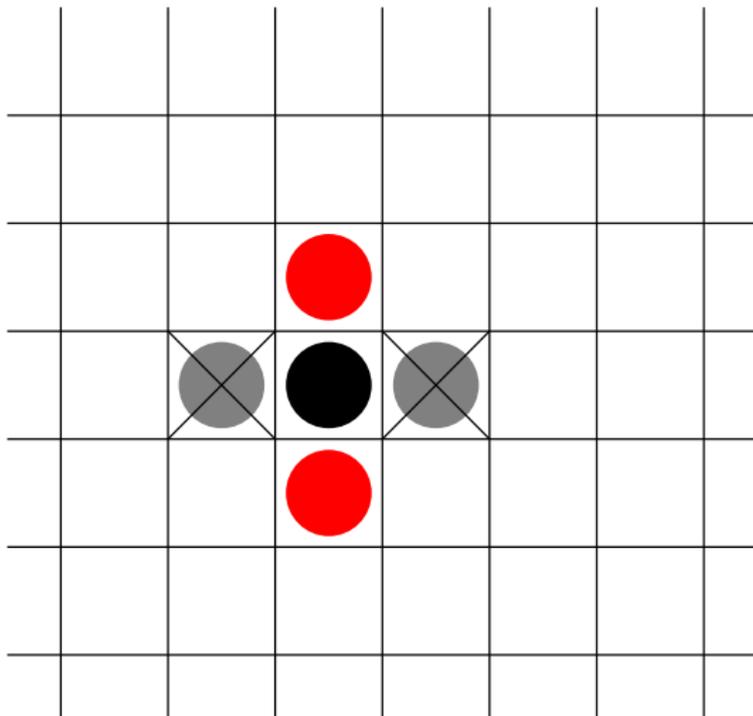
Oscillatori

Gli oscillatori sono configurazioni che si ripetono dopo un certo numero di passi. Il più semplice è il *lampeggiante*, di periodo 2, che è tra le forme di vita più comuni.



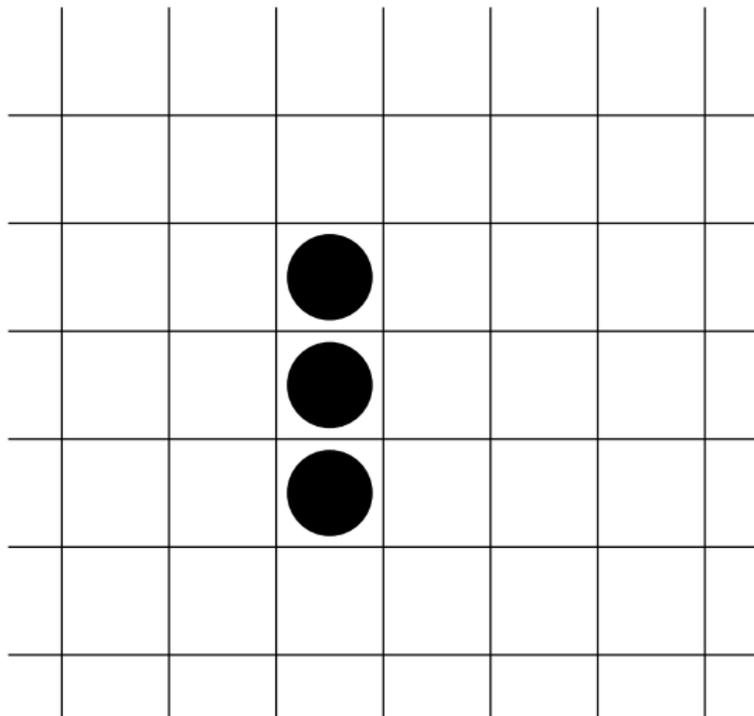
Oscillatori

Gli oscillatori sono configurazioni che si ripetono dopo un certo numero di passi. Il più semplice è il *lampeggiante*, di periodo 2, che è tra le forme di vita più comuni.



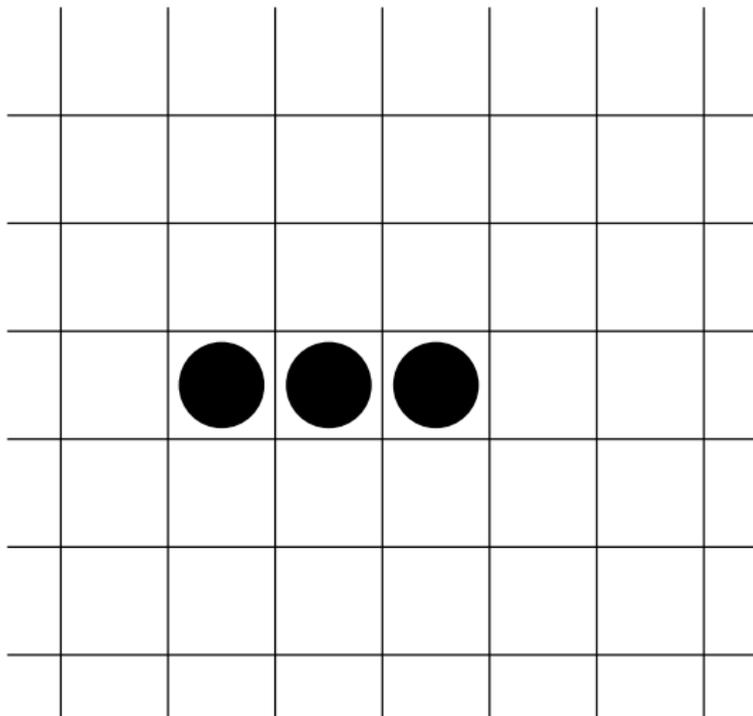
Oscillatori

Gli oscillatori sono configurazioni che si ripetono dopo un certo numero di passi. Il più semplice è il *lampeggiante*, di periodo 2, che è tra le forme di vita più comuni.



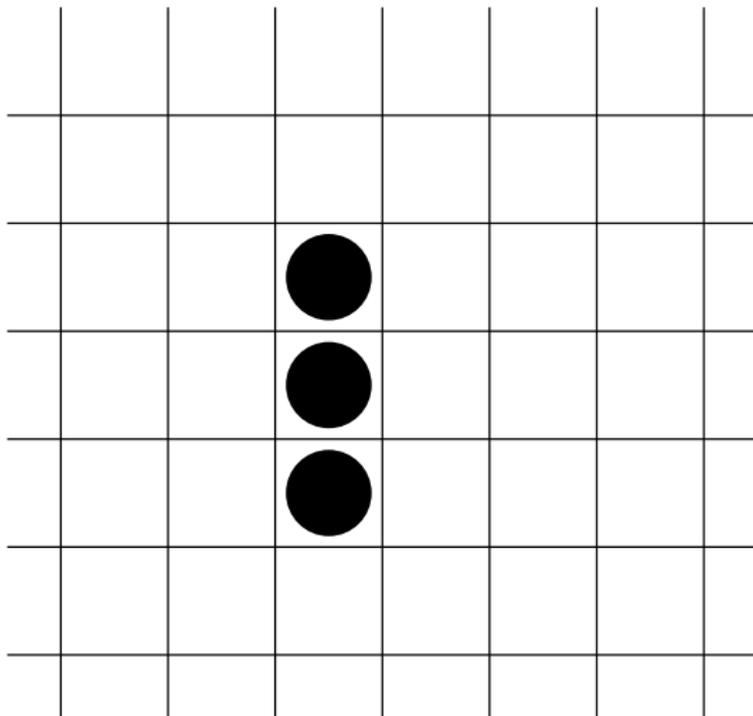
Oscillatori

Gli oscillatori sono configurazioni che si ripetono dopo un certo numero di passi. Il più semplice è il *lampeggiante*, di periodo 2, che è tra le forme di vita più comuni.



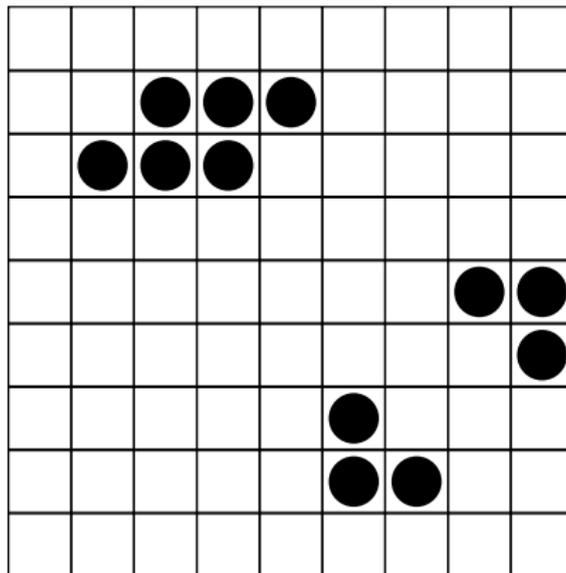
Oscillatori

Gli oscillatori sono configurazioni che si ripetono dopo un certo numero di passi. Il più semplice è il *lampeggiante*, di periodo 2, che è tra le forme di vita più comuni.



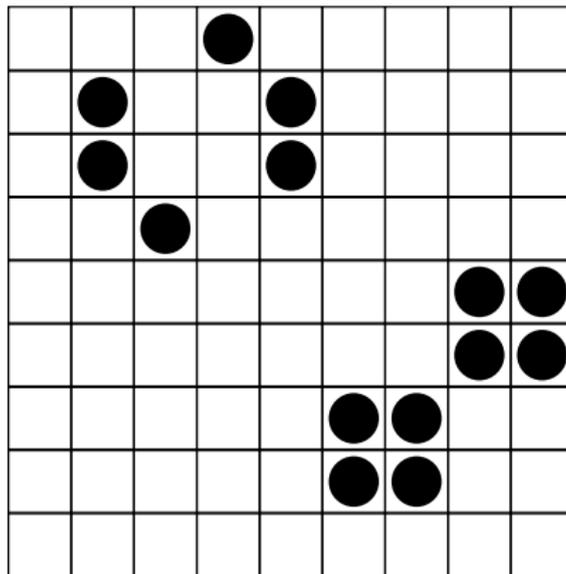
Oscillatori

Gli altri oscillatori di periodo 2 sono il *rospo* e il *bacon*.



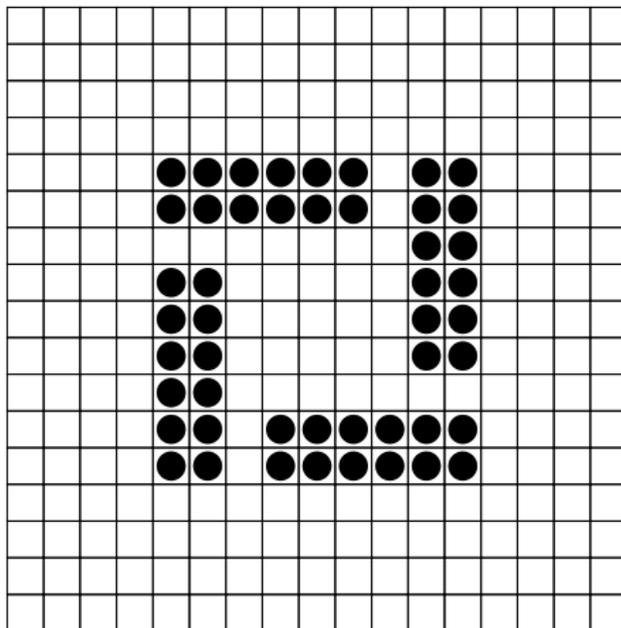
Oscillatori

Gli altri oscillatori di periodo 2 sono il *rospo* e il *bacon*.



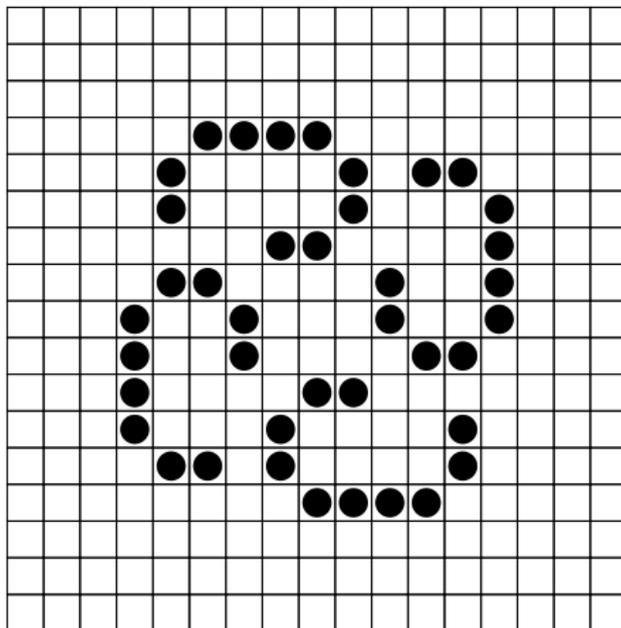
La galassia

La *galassia* è un oscillatore molto bello, di periodo 8.



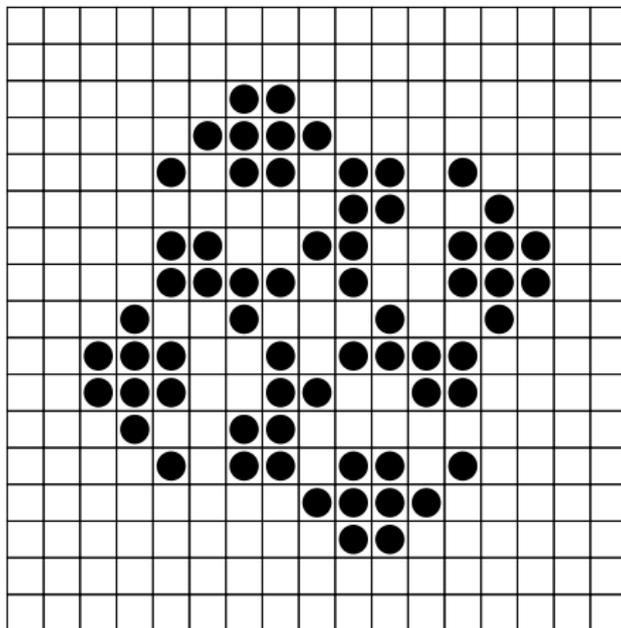
La galassia

La *galassia* è un oscillatore molto bello, di periodo 8.



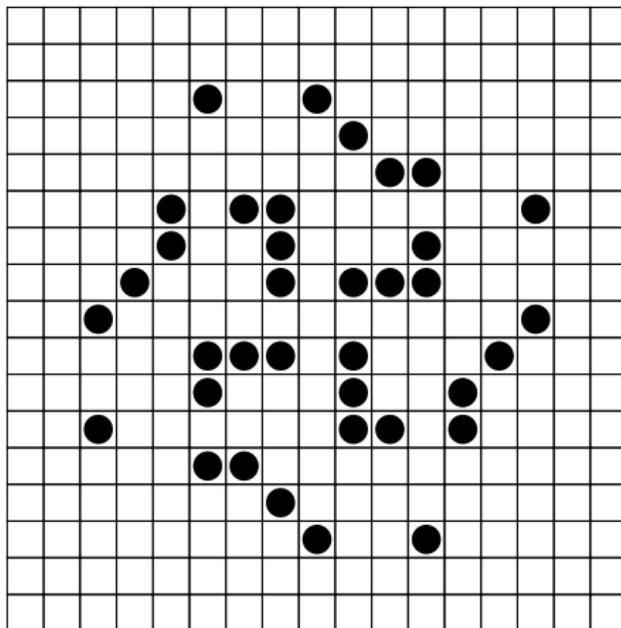
La galassia

La *galassia* è un oscillatore molto bello, di periodo 8.



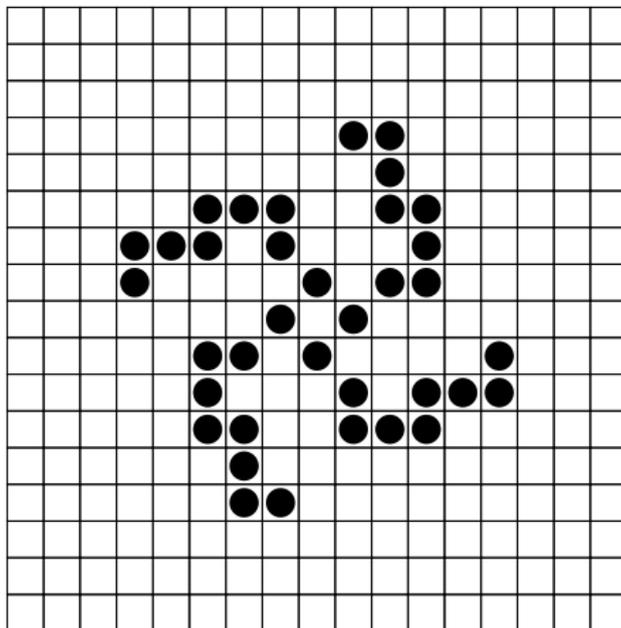
La galassia

La *galassia* è un oscillatore molto bello, di periodo 8.



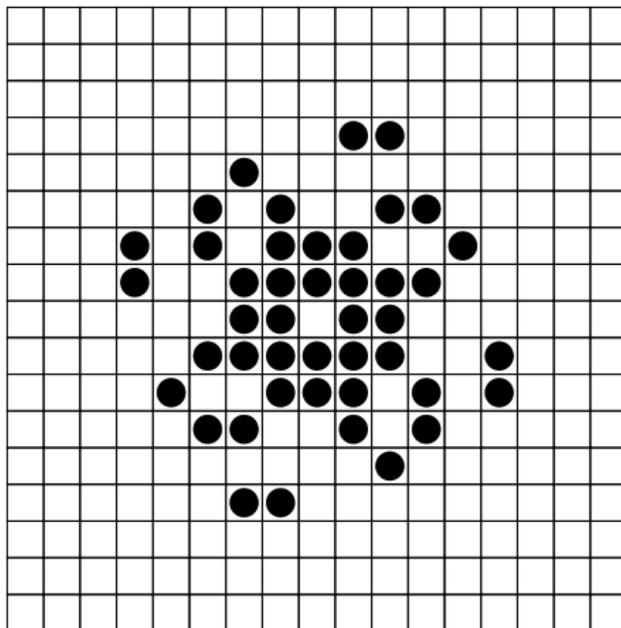
La galassia

La *galassia* è un oscillatore molto bello, di periodo 8.



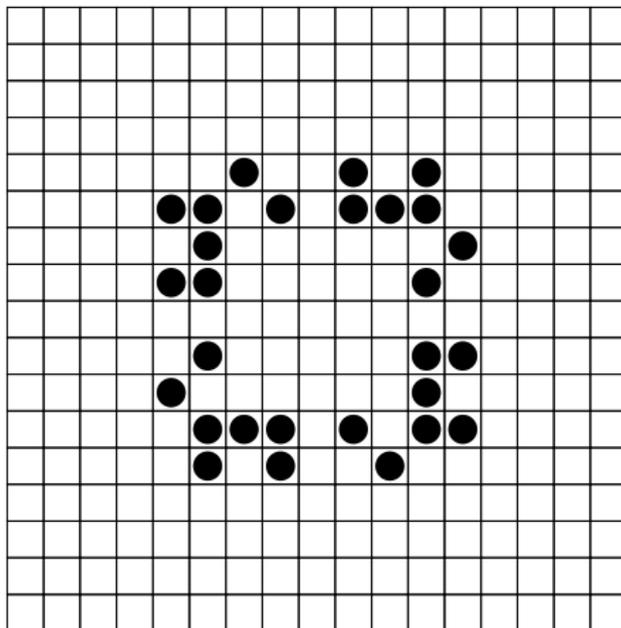
La galassia

La *galassia* è un oscillatore molto bello, di periodo 8.



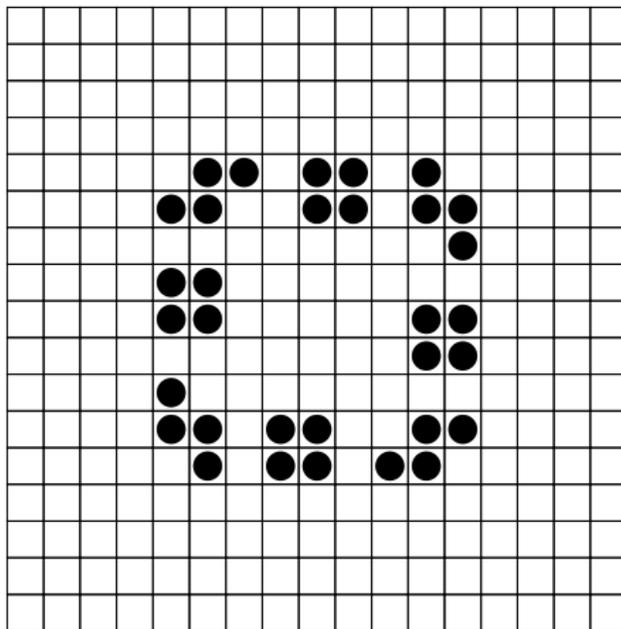
La galassia

La *galassia* è un oscillatore molto bello, di periodo 8.



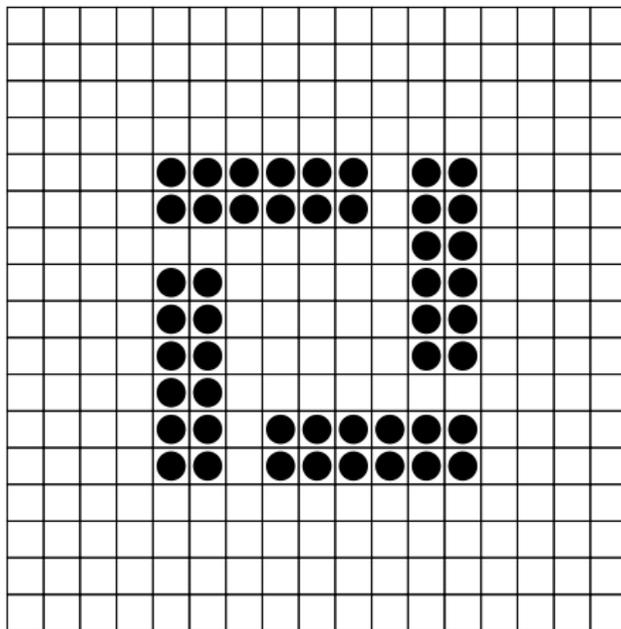
La galassia

La *galassia* è un oscillatore molto bello, di periodo 8.



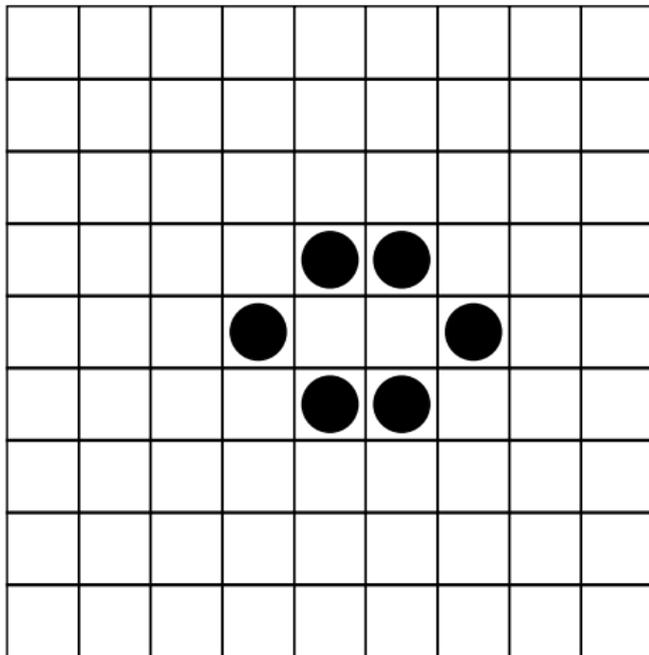
La galassia

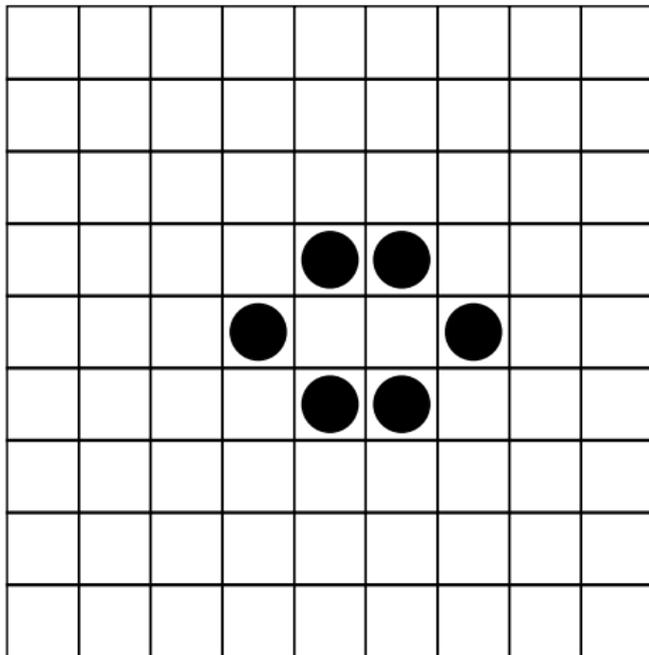
La *galassia* è un oscillatore molto bello, di periodo 8.



Alcune semplici configurazioni: fila da 4

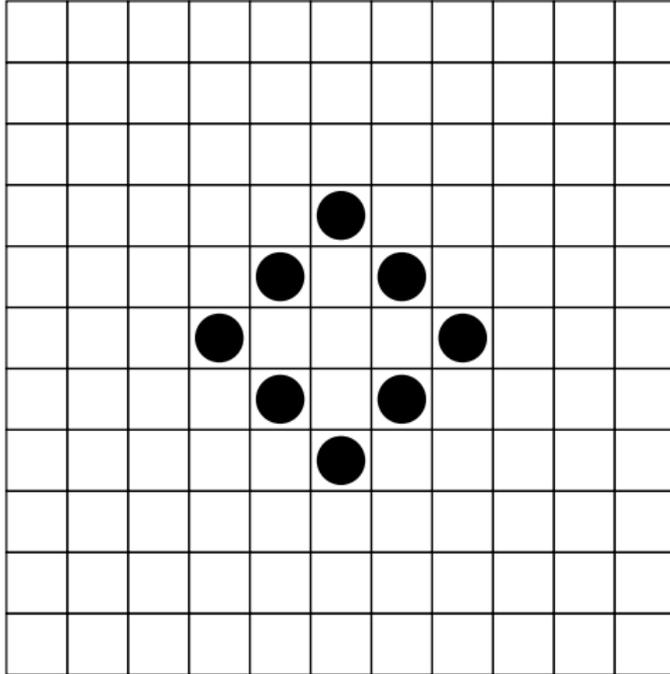
▶ Salta animazione

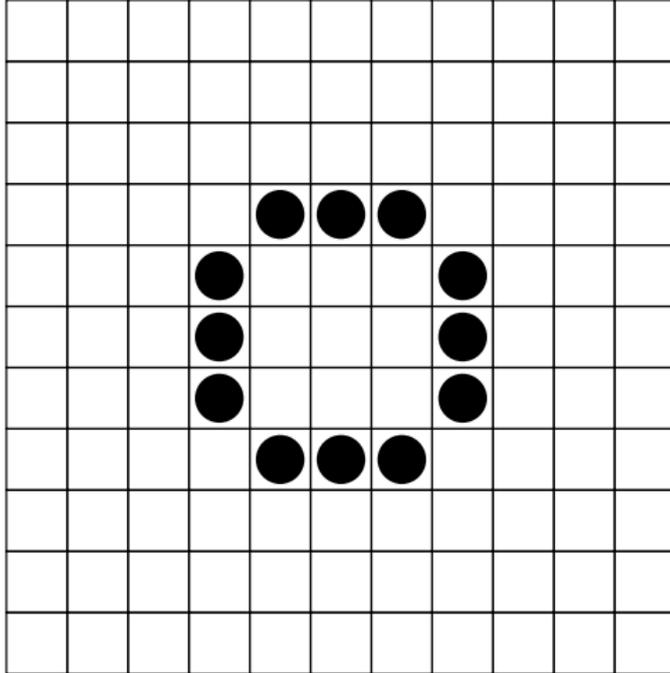


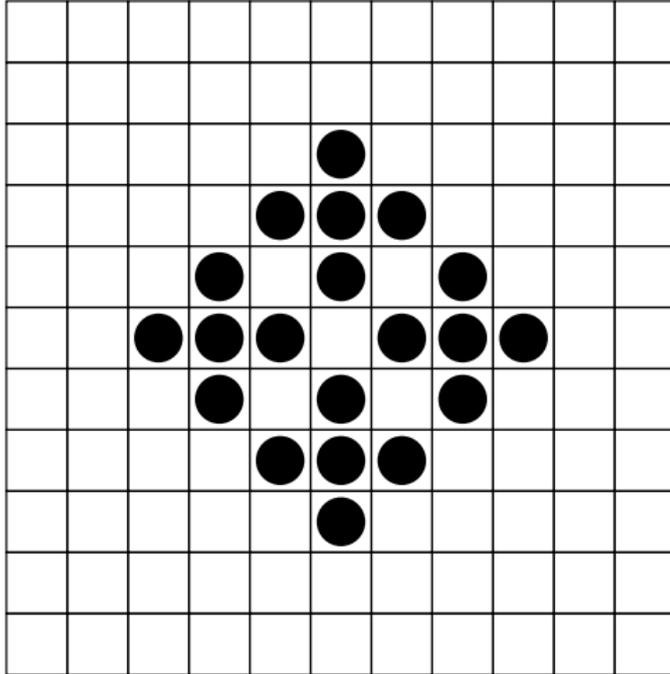


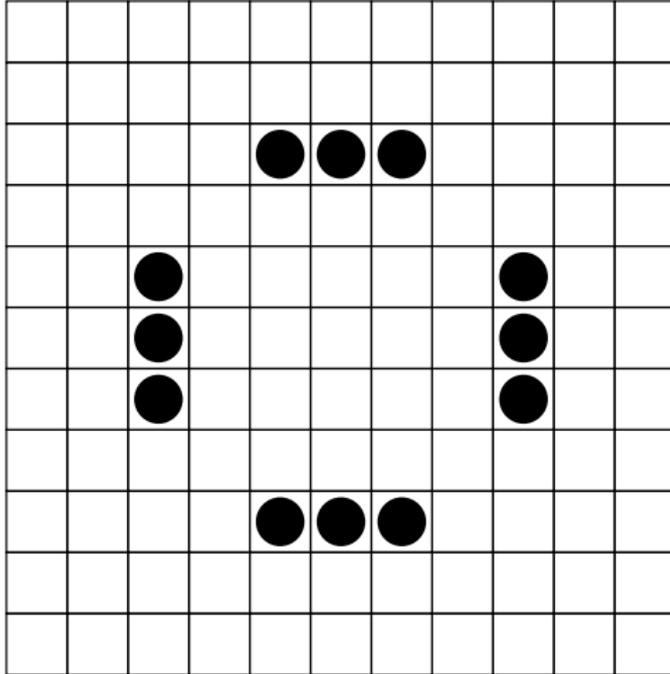
▶ Torna indietro

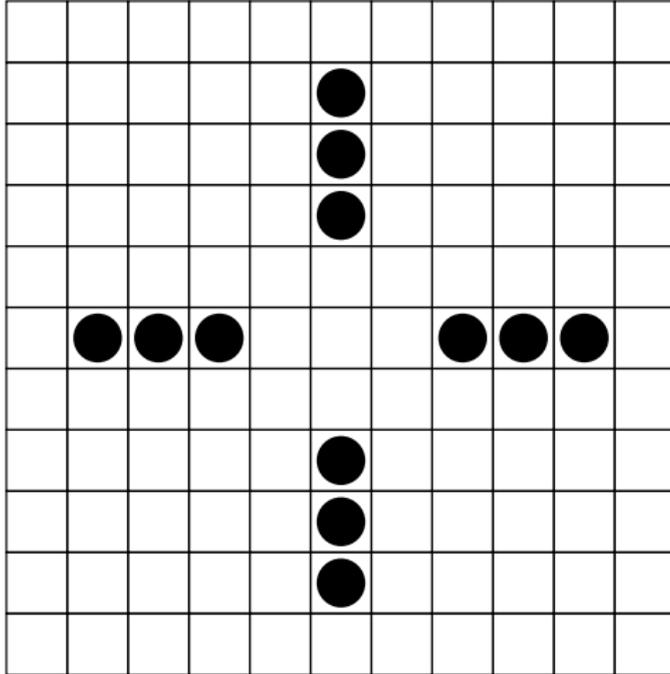
▶ Salta animazione

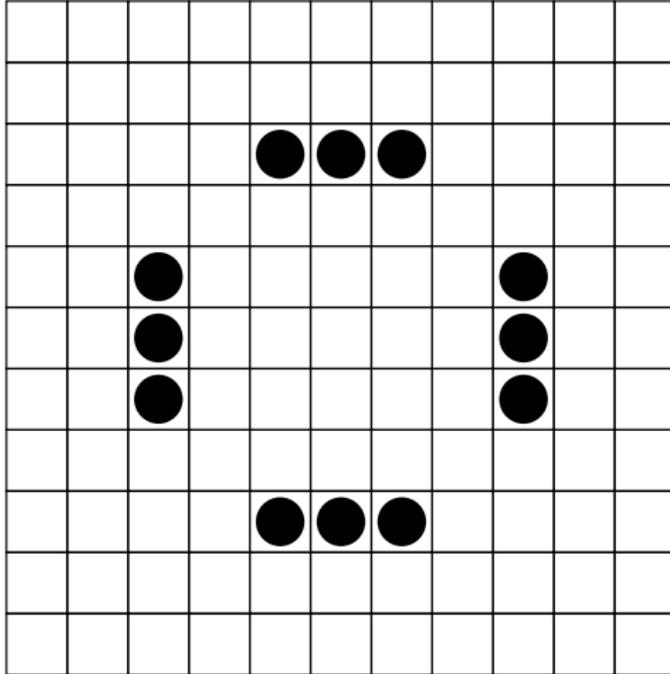






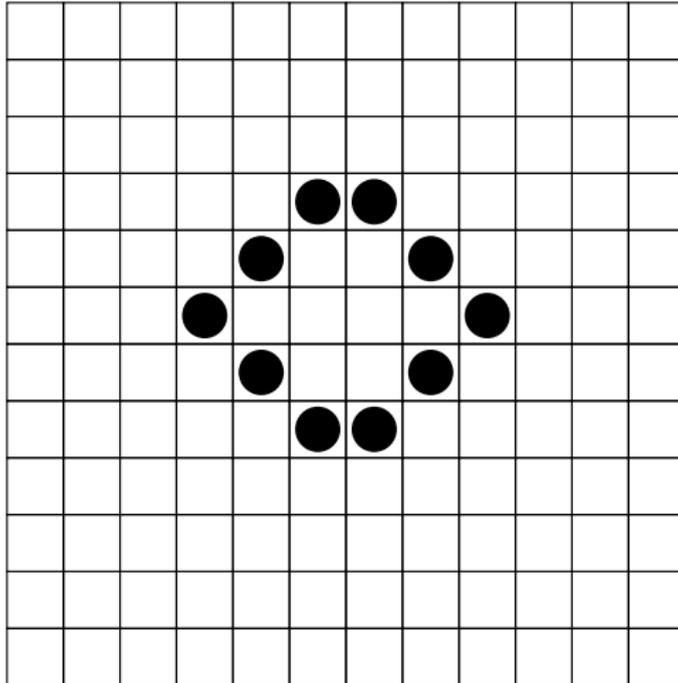


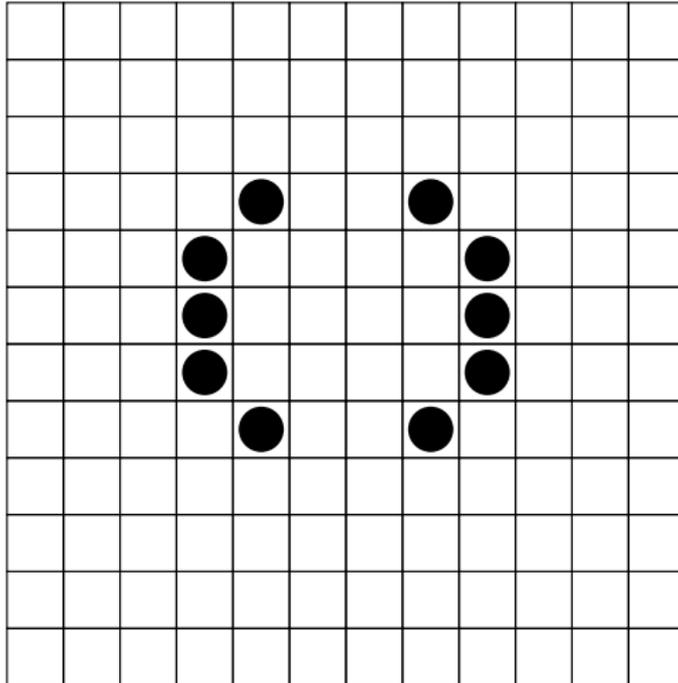


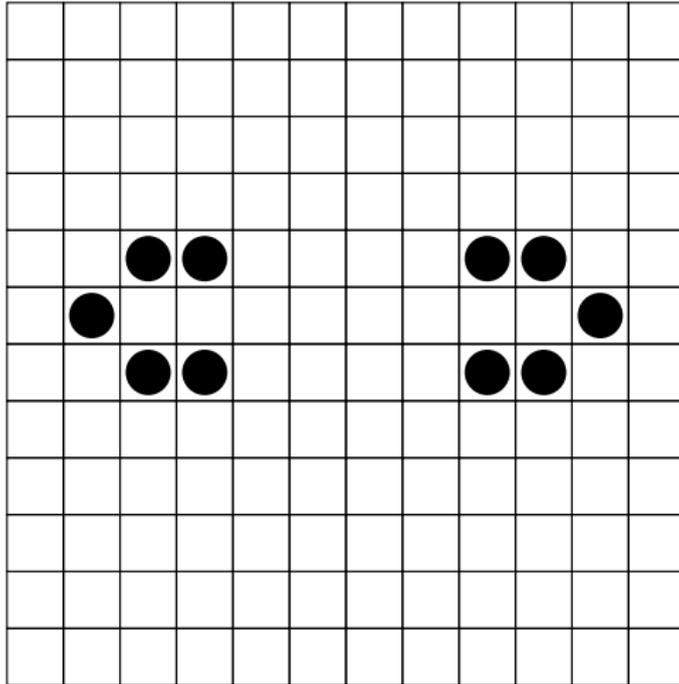


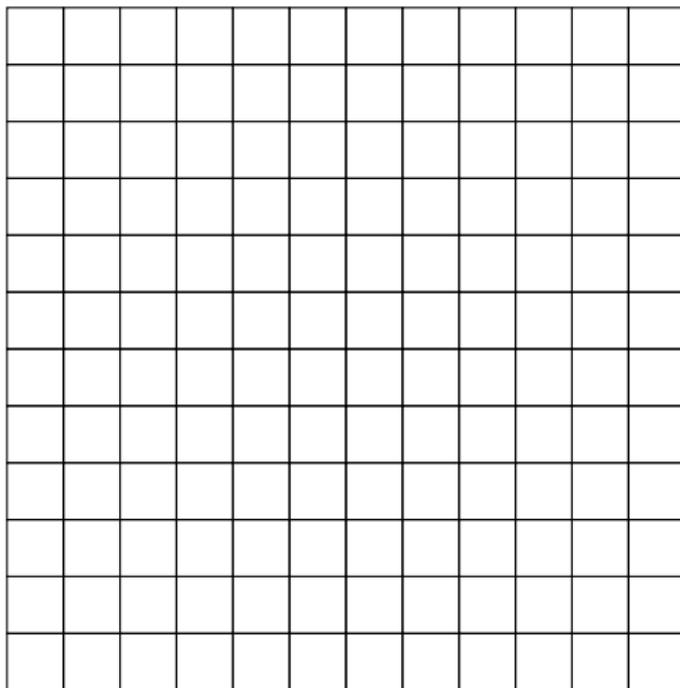
▶ Torna indietro

▶ Salta animazione



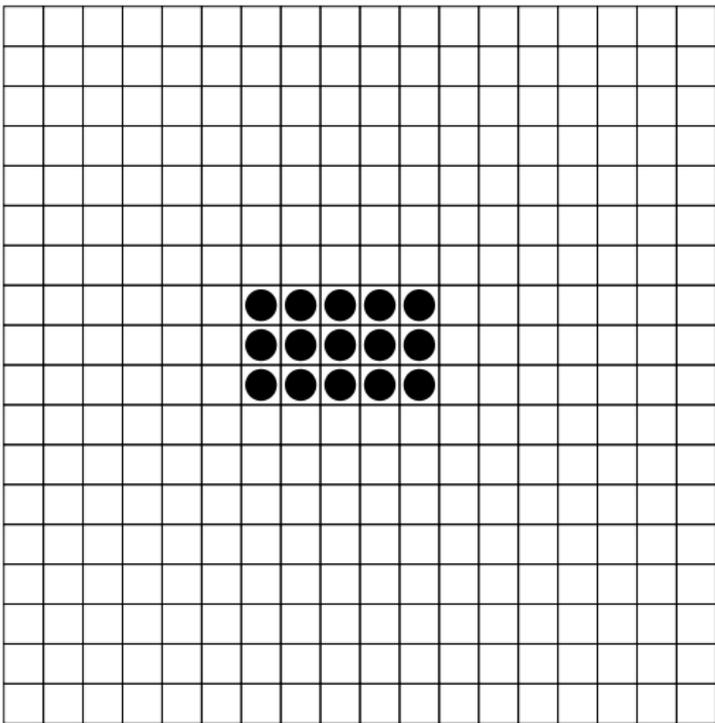


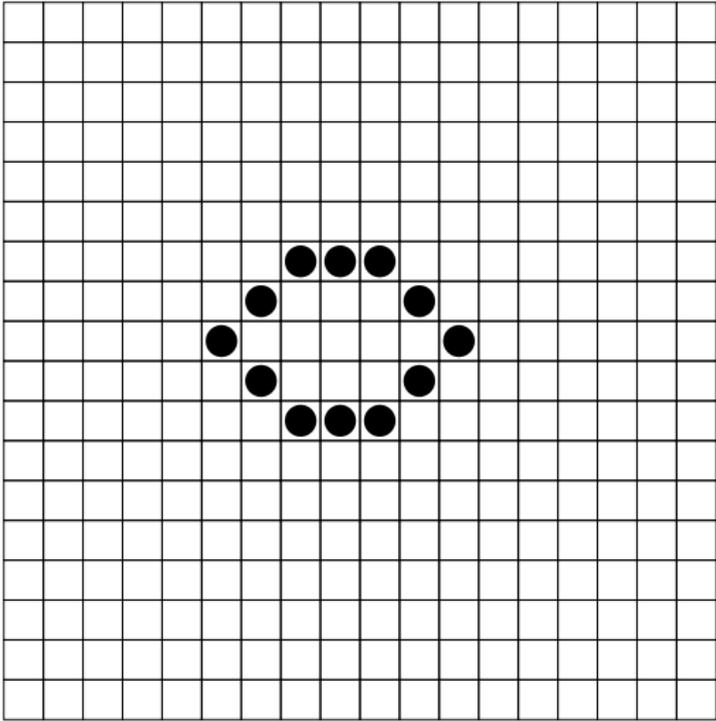


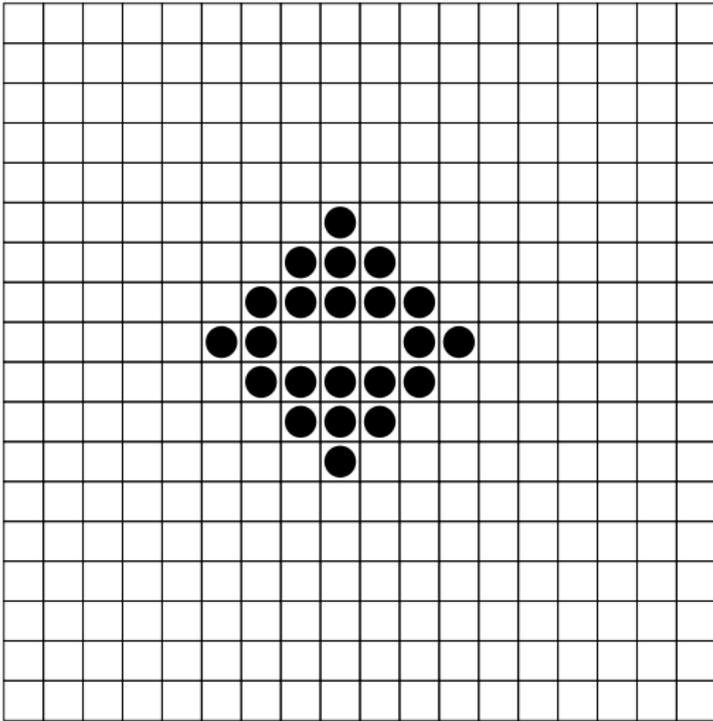


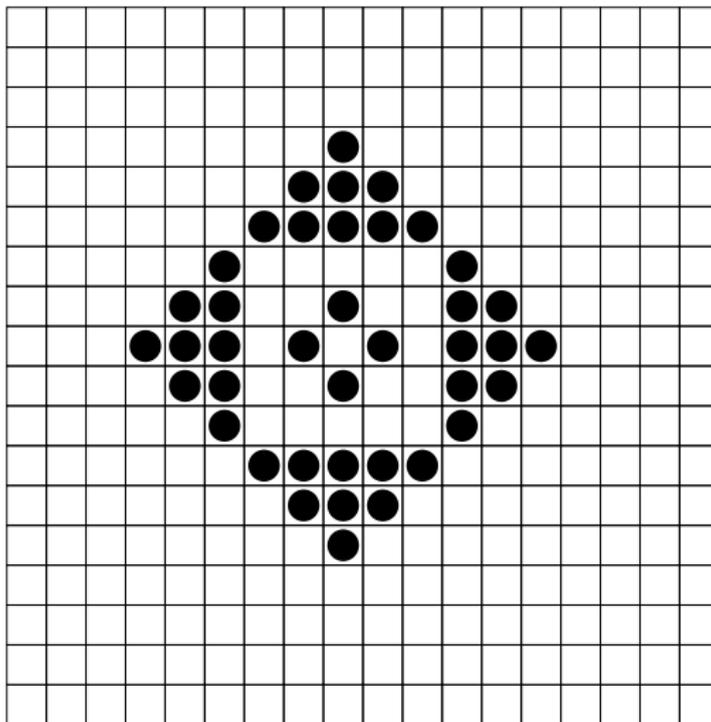
▶ Torna indietro

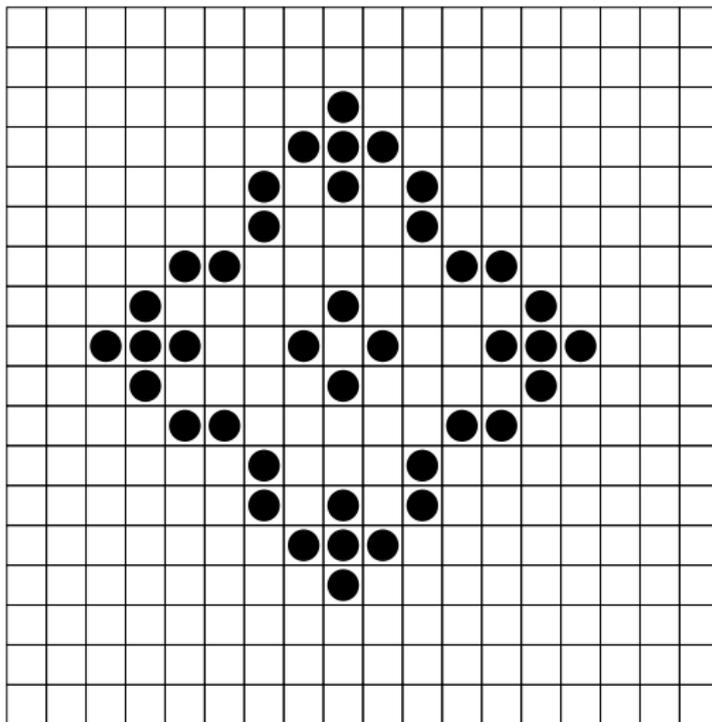
▶ Salta animazione

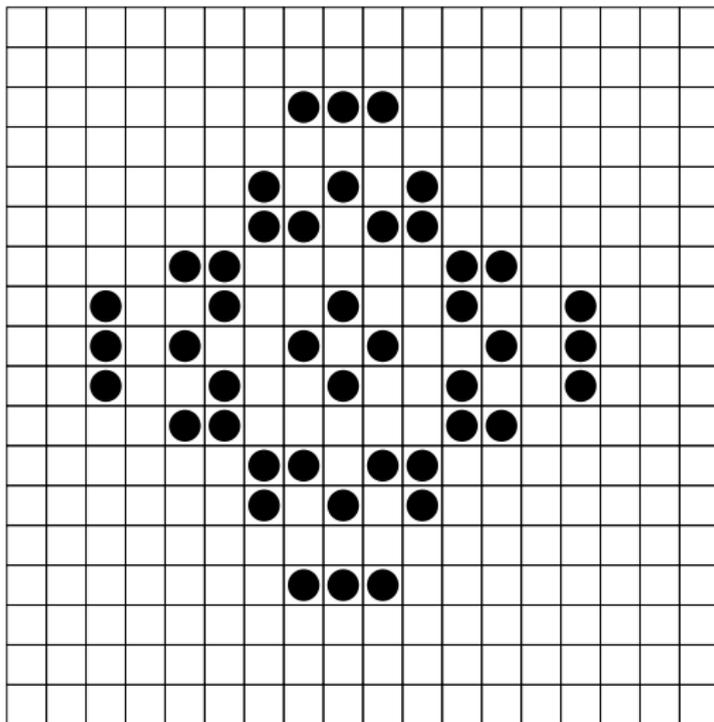


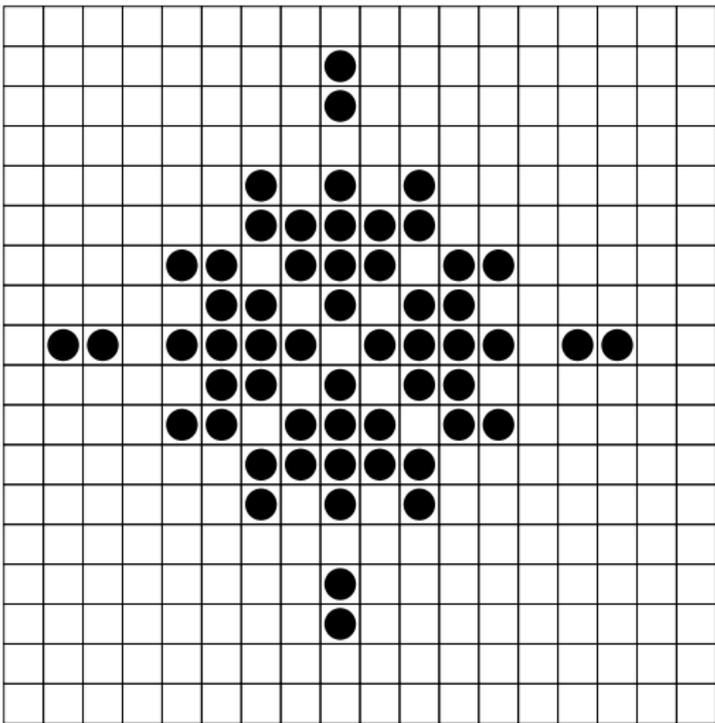


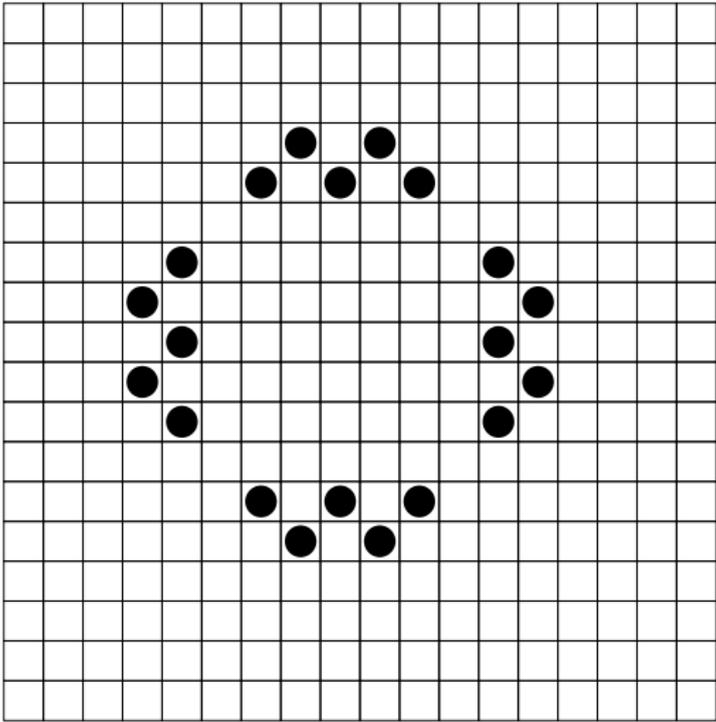












Fine animazione fila da 7

▶ Torna indietro

La classificazione è complessa

Riassumiamo che cosa succede partendo dalle configurazioni molto semplici fatte da una riga di n celle vive:

La classificazione è complessa

Riassumiamo che cosa succede partendo dalle configurazioni molto semplici fatte da una riga di n celle vive:

$n = 3$ lampeggiante	$n = 4$ alveare	$n = 5$ quattro lampeggianti
$n = 6$ scompare	$n = 7$ quattro alveari	$n = 8$ blocchi e alveari
$n = 9$ lampeggianti	$n = 10$ pentadecatlon	$n = 11$ due lampeggianti
$n = 12$ due alveari	$n = 13$ due lampeggianti	$n = 14$ scompare
$n = 15$ scompare	$n = 16$ lampeggianti	$n = 17$ quattro blocchi
$n = 18$ scompare	$n = 19$ scompare	$n = 20$ due blocchi

La classificazione è complessa

Riassumiamo che cosa succede partendo dalle configurazioni molto semplici fatte da una riga di n celle vive:

$n = 3$ lampeggiante	$n = 4$ alveare	$n = 5$ quattro lampeggianti
$n = 6$ scompare	$n = 7$ quattro alveari	$n = 8$ blocchi e alveari
$n = 9$ lampeggianti	$n = 10$ pentadecatlon	$n = 11$ due lampeggianti
$n = 12$ due alveari	$n = 13$ due lampeggianti	$n = 14$ scompare
$n = 15$ scompare	$n = 16$ lampeggianti	$n = 17$ quattro blocchi
$n = 18$ scompare	$n = 19$ scompare	$n = 20$ due blocchi

Life è un sistema non lineare!

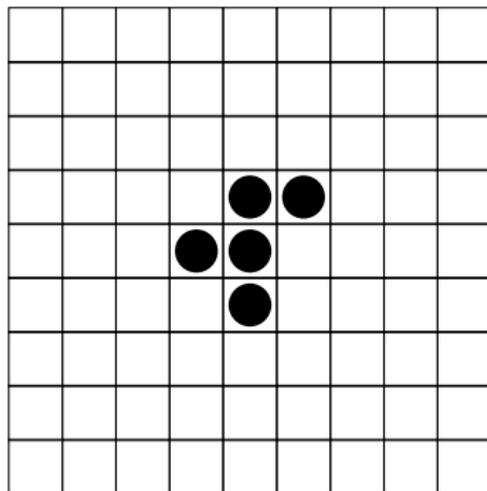
(non vale il principio di sovrapposizione degli effetti. . .)

Esempio: H e Π .

<http://golly.sourceforge.net>

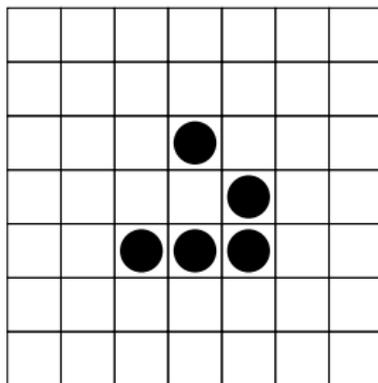
Per Linux, Windows, Mac, iPad, Android

Il pentamino R



Il pentamino R si stabilizza dopo 1103 generazioni, raggiungendo una popolazione di 116 cellule: 8 blocchi, 6 glider, 4 alveari, 4 lampeggianti, una barca, una pagnotta, una nave.

L'aliante e altre astronavi



In Life esistono oggetti che traslano indefinitamente! Tra questi il più famoso è il *glider* (aliante), scoperto da Conway, e tre oggetti chiamati *astronave leggera*, *astronave media*, *astronave pesante*.

Altri esempi

gospers

proliferazione

Numeri *primi*

Glider *synthesis*

Altri esempi

gosper

proliferazione

Numeri *primi*

Glider *synthesis*

Nel 1982 Conway dimostrò che Life è una *macchina di Turing universale*, ovvero ha le stesse capacità di un qualsiasi computer!

Velocità e direzione delle astronavi

La velocità massima consentita in Life è di una casella alla volta, e viene denotata con c . Si dimostra che non possono esistere oggetti che si muovono con una velocità superiore a $c/2$.

Le tre astronavi viste prima hanno proprio velocità $c/2$. Il glider invece ha velocità $c/4$ (si conta solo lo spostamento orizzontale o verticale).

Velocità e direzione delle astronavi

La velocità massima consentita in Life è di una casella alla volta, e viene denotata con c . Si dimostra che non possono esistere oggetti che si muovono con una velocità superiore a $c/2$.

Le tre astronavi viste prima hanno proprio velocità $c/2$. Il glider invece ha velocità $c/4$ (si conta solo lo spostamento orizzontale o verticale).

Fino al 2010 tutte le astronavi conosciute si muovevano ortogonalmente (come l'astronave leggera) o a 45 gradi (come il glider). Solo nel 2010 Andrew Wade ha scoperto (o inventato?) Gemini, formata da 846278 cellule, che si sposta di 5120 celle in verticale e 1024 in orizzontale ogni 33699586 generazioni, quindi si muove con una pendenza di 5.

Velocità e direzione delle astronavi

La velocità massima consentita in Life è di una casella alla volta, e viene denotata con c . Si dimostra che non possono esistere oggetti che si muovono con una velocità superiore a $c/2$.

Le tre astronavi viste prima hanno proprio velocità $c/2$. Il glider invece ha velocità $c/4$ (si conta solo lo spostamento orizzontale o verticale).

Fino al 2010 tutte le astronavi conosciute si muovevano ortogonalmente (come l'aeronave leggera) o a 45 gradi (come il glider). Solo nel 2010 Andrew Wade ha scoperto (o inventato?) Gemini, formata da 846278 cellule, che si sposta di 5120 celle in verticale e 1024 in orizzontale ogni 33699586 generazioni, quindi si muove con una pendenza di 5.

A luglio del 2014 è stata costruita una nuova astronave, la Parallel HBK, molto più semplice di Gemini (132945 cellule), che si muove con pendenza 2 in 245912 generazioni, grazie al lavoro di Chris Cain, Ivan Fomichev, Dave Greene e Adam Goucher, che hanno collaborato lavorando su un blog.

[parallel_HBK](#): $(6,3)c/245912$

Velocità e direzione delle astronavi – 2

Il 28 dicembre 2014 Brett Berger ha costruito Waterbear, un'astronave che percorre 23 celle in verticale e 5 in orizzontale ogni 79 generazioni. Pur essendo formata da 197896 cellule, che sono più di quelle della Parallel KBH, ha una velocità estremamente più alta:

waterbear: $(23,5)c/79$

Life simula se stesso

Outer Totalistic Cellular Automata Meta Pixel (OTCAMP) è una configurazione di Life che occupa un'area di 2048×2048 celle e ha una periodicità di 35328 generazioni.

Life simula se stesso

Outer Totalistic Cellular Automata Meta Pixel (OTCAMP) è una configurazione di Life che occupa un'area di 2048x2048 celle e ha una periodicità di 35328 generazioni.

Ogni cellula “piena” è fatta da più di 60000 cellule vive, mentre le cellule “vuote” da circa 23000.

Life simula se stesso

Outer Totalistic Cellular Automata Meta Pixel (OTCAMP) è una configurazione di Life che occupa un'area di 2048x2048 celle e ha una periodicità di 35328 generazioni.

Ogni cellula “piena” è fatta da più di 60000 cellule vive, mentre le cellule “vuote” da circa 23000.

Con queste “celle” si può emulare Life dentro se stesso. E si possono anche emulare altre regole dentro Life. . .

Life simula se stesso

Outer Totalistic Cellular Automata Meta Pixel (OTCAMP) è una configurazione di Life che occupa un'area di 2048x2048 celle e ha una periodicità di 35328 generazioni.

Ogni cellula “piena” è fatta da più di 60000 cellule vive, mentre le cellule “vuote” da circa 23000.

Con queste “celle” si può emulare Life dentro se stesso. E si possono anche emulare altre regole dentro Life. . .

metapixel

meta-blinker

meta-tennis

replicator-metapixel

contatore

Formalizziamo il modello

Per definire un *automa cellulare* abbiamo bisogno di:

- un *ambiente* in cui mettere le cellule
- un insieme di *stati* della cellula
- la definizione di *intorno* di una cellula
- una *regola* di evoluzione

Formalizziamo il modello

Per definire un *automa cellulare* abbiamo bisogno di:

- un *ambiente* in cui mettere le cellule
- un insieme di *stati* della cellula
- la definizione di *intorno* di una cellula
- una *regola* di evoluzione

Nel caso di Life abbiamo:

- griglia quadrata infinita $\mathbb{Z} \times \mathbb{Z}$
- cellula viva $s = 1$
cellula morta $s = 0$
- l'intorno $U(x)$ dato dalle otto caselle vicine (senza x)
- la regola di Life è data da:

Formalizziamo il modello

Per definire un *automa cellulare* abbiamo bisogno di:

- un *ambiente* in cui mettere le cellule
- un insieme di *stati* della cellula
- la definizione di *intorno* di una cellula
- una *regola* di evoluzione

$$s_{t+1}(x) = \begin{cases} 1 & \text{se } p_t(x) = 3 \\ 1 & \text{se } p_t(x) = 2 \text{ e } s_t(x) = 1 \\ 0 & \text{altrimenti} \end{cases}$$

dove $p_t(x)$ è il numero di cellule vive in $U(x)$ al tempo t , cioè

$$p_t(x) = \sum_{y \in U(x)} s_t(y)$$

Nel caso di Life abbiamo:

- griglia quadrata infinita $\mathbb{Z} \times \mathbb{Z}$
- cellula viva $s = 1$
cellula morta $s = 0$
- l'intorno $U(x)$ dato dalle otto caselle vicine (senza x)
- la regola di Life è data da:

Generalizzazioni

- griglie diverse (triangolari, esagonali, toroidali, tridimensionali, Penrose, casuali. . .)

Generalizzazioni

- griglie diverse (triangolari, esagonali, toroidali, tridimensionali, Penrose, casuali. . .)
- più di due stati

Generalizzazioni

- griglie diverse (triangolari, esagonali, toroidali, tridimensionali, Penrose, casuali. . .)
- più di due stati
- altri intorni (Von Neumann)

Generalizzazioni

- griglie diverse (triangolari, esagonali, toroidali, tridimensionali, Penrose, casuali. . .)
- più di due stati
- altri intorni (Von Neumann)
- regole probabilistiche

Generalizzazioni

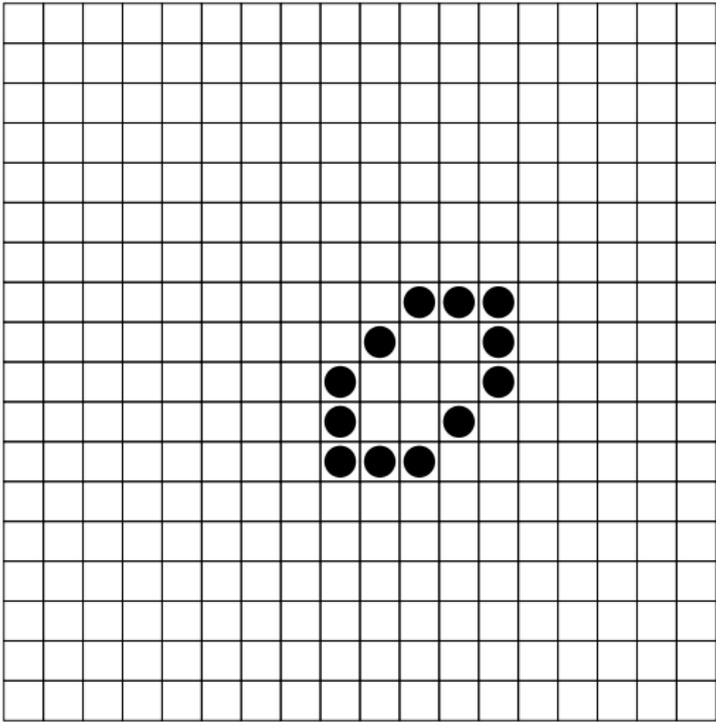
- griglie diverse (triangolari, esagonali, toroidali, tridimensionali, Penrose, casuali. . .)
- più di due stati
- altri intorni (Von Neumann)
- regole probabilistiche
- regole asincrone (gli stati cambiano uno alla volta)

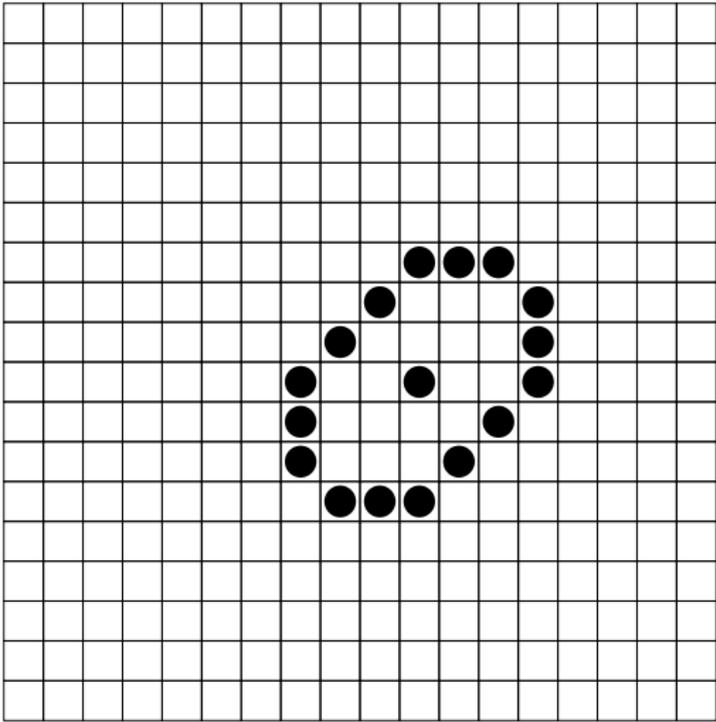
Un'altra regola molto studiata è la B36/S23. Conway stesso afferma:
It seems to me that B36/S23 is really the game I should have found, since it's so rich in nice things.

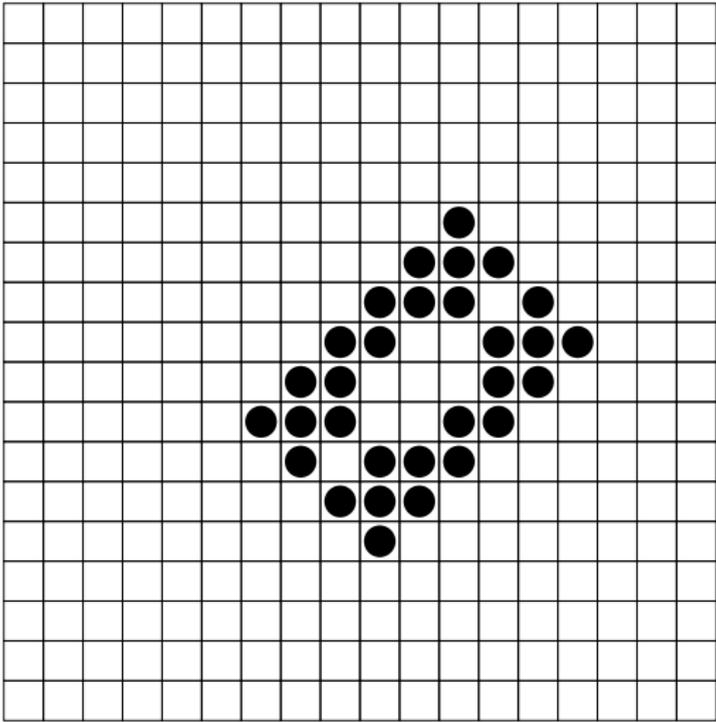
Un'altra regola molto studiata è la B36/S23. Conway stesso afferma:
It seems to me that B36/S23 is really the game I should have found, since it's so rich in nice things.

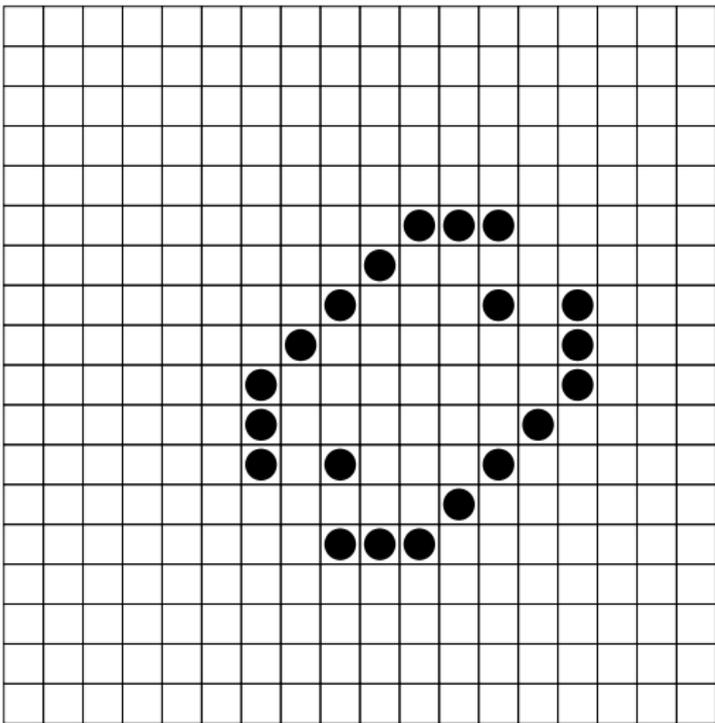
L'interesse principale di questa regola è che ammette un semplice *riproduttore*, cioè una forma che si riproduce identica a se stessa.

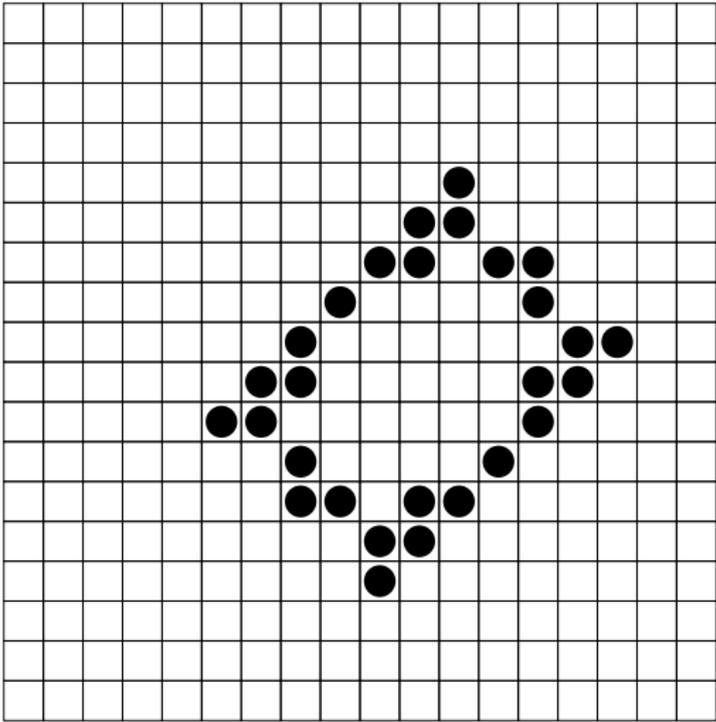
▶ Salta animazione

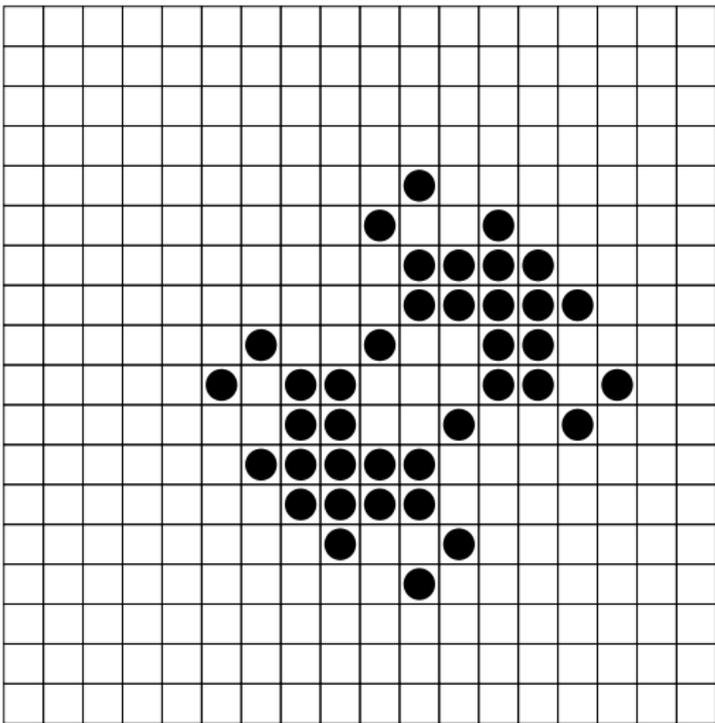


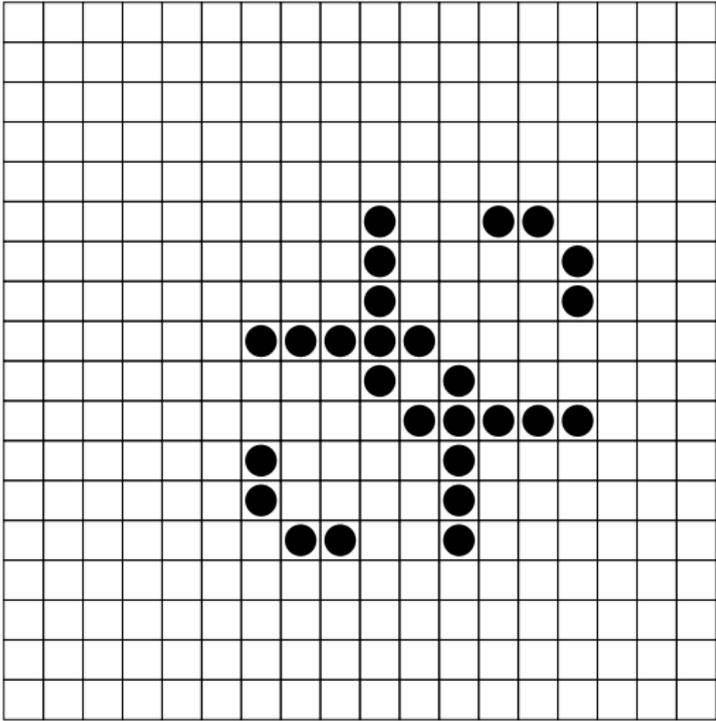


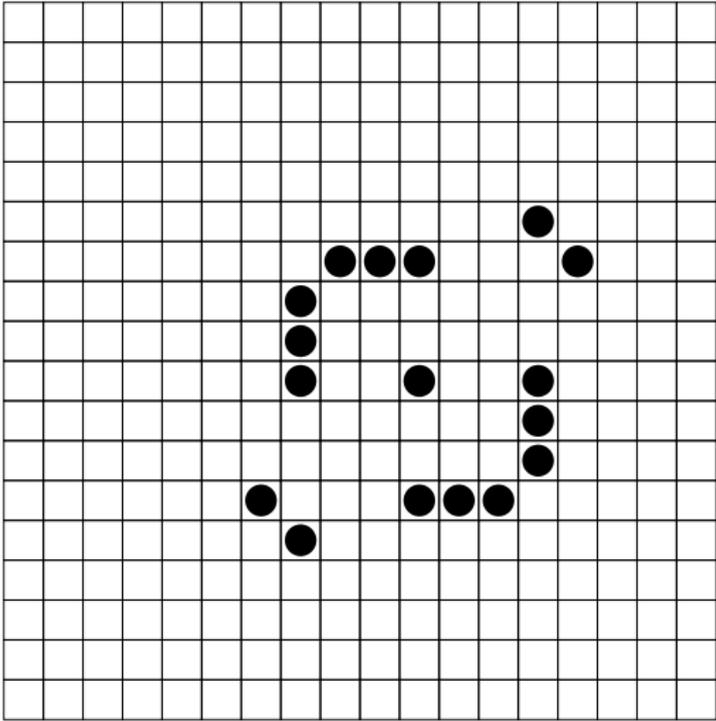


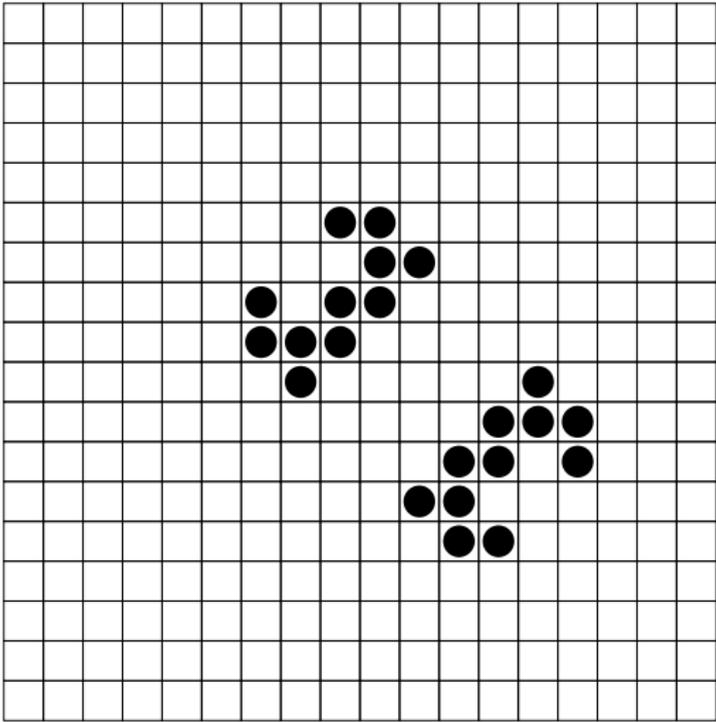












Fine animazione riproduttore

▶ Torna indietro

Esiste anche una regola in cui *ogni* forma si autoreplica: B1357/S1357.

Ovviamente questa regola non dà luogo a sviluppi interessanti, anche se può essere simpatica da vedere.

esempio

Riproduttore in Life

Poiché Life è una macchina di Turing universale, questo implica che anche in Life esiste un riproduttore. Una tale configurazione però non era mai stata trovata, fino al novembre del 2013, quando Dave Greene, basandosi sulle idee dell'astronave Gemini, è riuscito a costruire i primi riproduttori di Life.

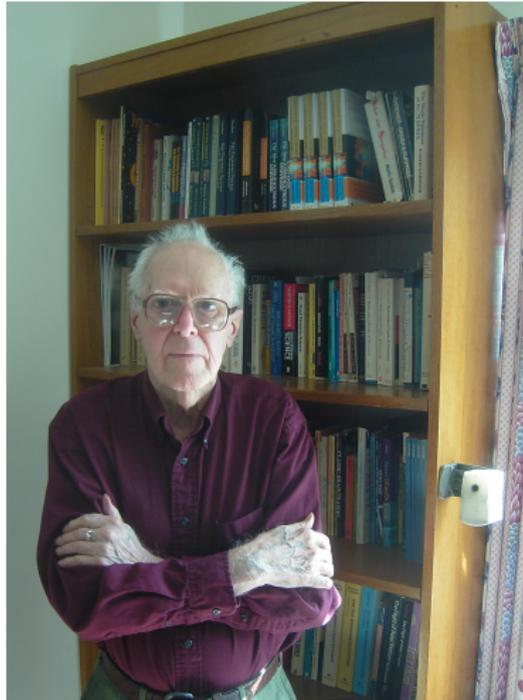
Riproduttore in Life

Poiché Life è una macchina di Turing universale, questo implica che anche in Life esiste un riproduttore. Una tale configurazione però non era mai stata trovata, fino al novembre del 2013, quando Dave Greene, basandosi sulle idee dell'astronave Gemini, è riuscito a costruire i primi riproduttori di Life.

I più veloci impiegano circa 88 milioni di passi per riprodursi. Nell'esempio ne vedremo uno che ne impiega 237 milioni, ma che è più veloce in Golly.

riproduttore

Per finire



gardner