Gödel
Mitos y realidades

Marco Aurelio Alzate Monroy (*)

Universidad Distrital Francisco José de Caldas

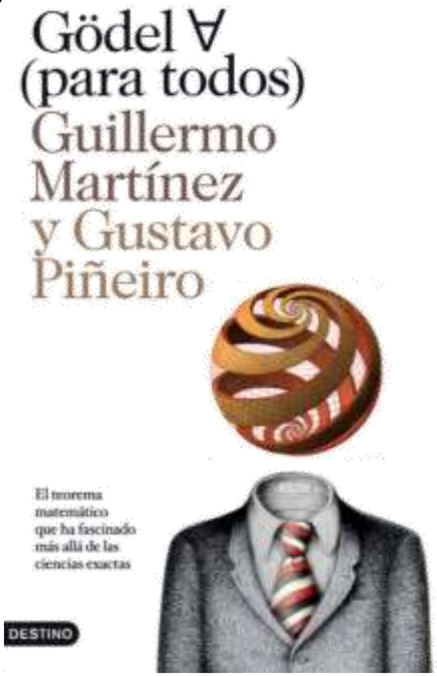
(*) No es un matemático. Es simplemente un ingeniero al que le dio por estudiar sistemas complejos… ¡Sean indulgentes, por favor!

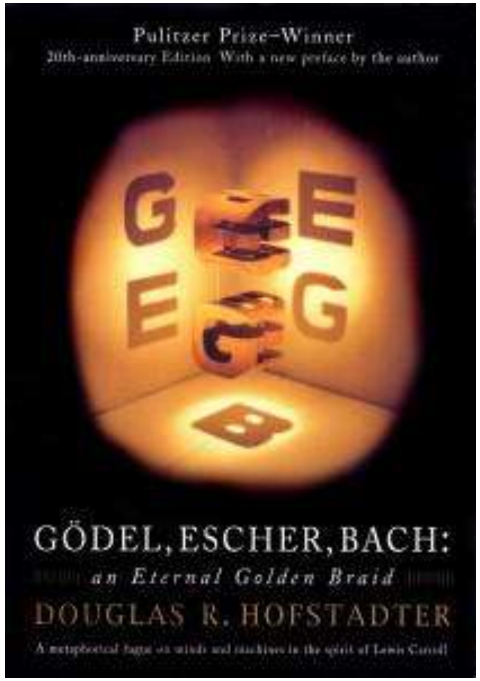
UNIVERSIDAD DISTRITAL FRANCISCO JOSE DE CALDAS

ideas
Investigación, desarrollo y aplicaciones en señales

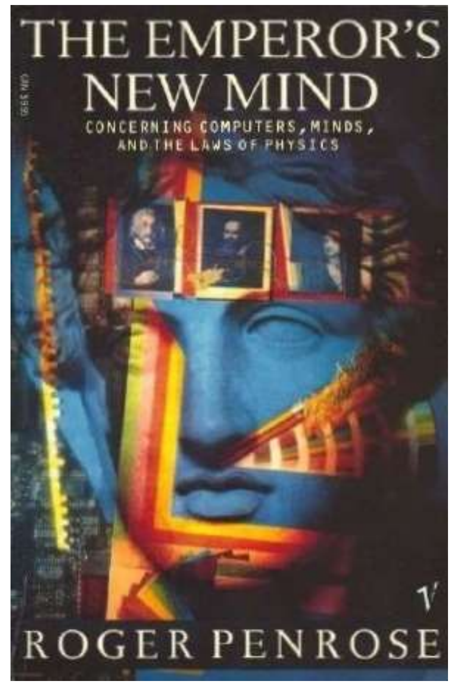
GITUD
Grupo de Investigación en Telecomunicaciones de la Universidad Distrital

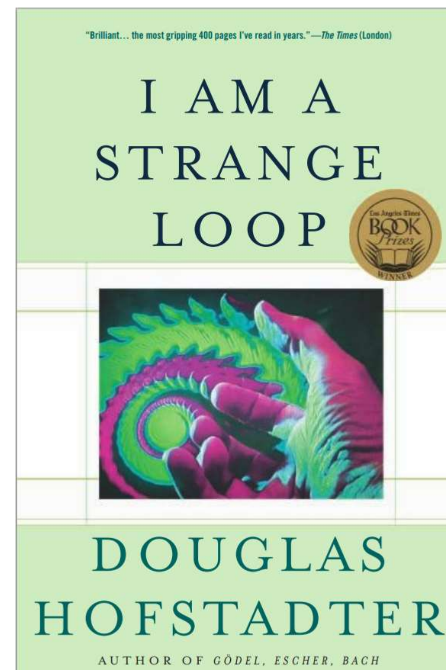
Complejidad UD





Los libros que quise estudiar para hoy…





…pero apenas alcancé a hojearlos

Algunos Mitos y Realidades en la Teoría de Sistemas Complejos

- Realidad: Los fenómenos complejos se dan lejos del equilibrio termodinámico, el equilibrio más estable. Mito: Los fenómenos complejos se dan lejos de cualquier equilibrio, en particular del equilibrio estable.
- Realidad: Los sistemas dinámicos lineales no pueden presentar comportamientos complejos. Mito: Los sistemas complejos son ajenos a cualquier forma de linealidad.
- Realidad: Cualquier forma de control centralizado va en contravía de la complejidad. Mito: Cualquier forma de control va en contravía de la complejidad.
- Realidad: Los sistemas vivos superan la capacidad computacional de la máquina de Turing, la cual no resuelve eficientemente los problemas NP. Mito: La vida es un problema NP.

Algunos Mitos con respecto a los teoremas de Gödel

- Mito: Los teoremas de Gödel establecen un límite a las pretensiones de la razón humana.
- Mito: Los teoremas de Gödel dicen que ninguna verdad puede establecerse de manera definitiva
- Mito: Los teoremas de Gödel dicen que, ni siquiera en las matemáticas, puede haber total certidumbre.
- Mito: El teorema de la consistencia de Gödel dice que ninguna teoría puede ser consistente.
- Mito: Los teoremas de Gödel dicen que ninguna teoría puede ser a la vez consistente y completa.
- Mito: Los teoremas de Gödel dicen que toda teoría de la aritmética es incompleta.
- Mito: Los teoremas de Gödel dicen que toda teoría recursiva es incompleta.
- Mito: Los teoremas de Gödel no tienen ninguna incidencia en las matemáticas.

Al final de la presentación veremos las realidades correspondientes

¿1+1=2?

¿La virgen María fue concebida sin pecado?

- $\forall x\, (x + 0 = x) \quad \Rightarrow \quad 1 + 0 = 1$
- $\forall x \forall y\, (\, x + S(y) = S(x + y)\,) \Rightarrow 1 + S(0) = S(1{+}0) = S(1)$
- $S(0) = 1$ y $S(1) = 2$
- $\Rightarrow 1 + 1 = 2$

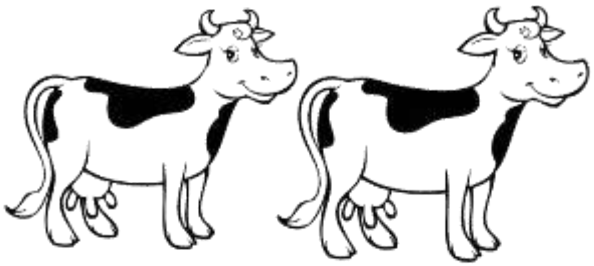
QED □

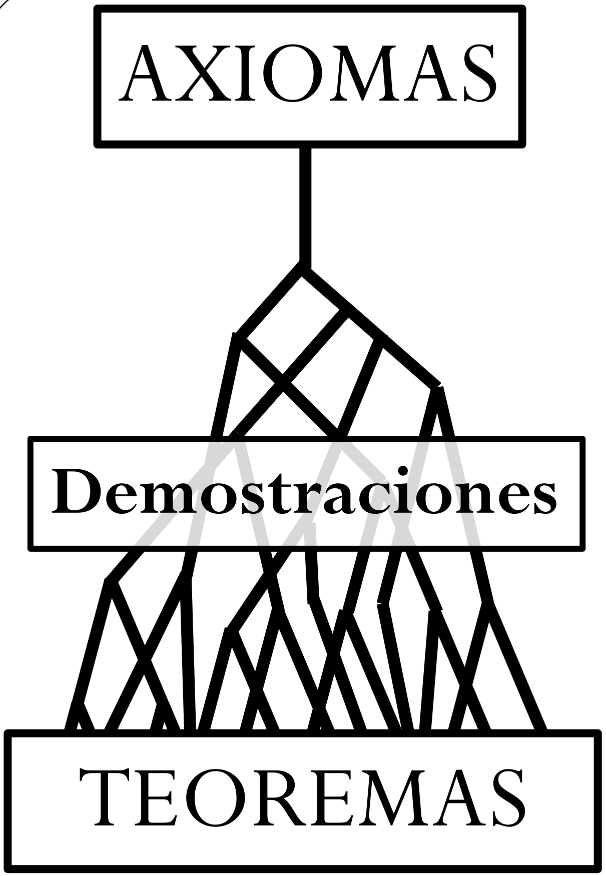
Quod erat demonstrandum

Una verdad

Demostrable

¡Isomorfismo!





Axiomas: Conjunto reducido de enunciados verdaderos, seleccionados de manera tal que puedan obtenerse de ellos, mediante demostración, todos los enunciados verdaderos en el sistema.

Demostración: Sucesión de enunciados, encadenados por pasos elementales estrictamente lógicos, de manera que cualquier persona (o máquina) pueda examinarlos para tener la seguridad de que no se ha cometido ningún error.

Teorema: Último enunciado de la demostración. Cuando el razonamiento es profundo, el teorema sorprende, maravilla: La creatividad, la inspiración y el arte participaron en la elección de los pasos que, entre todas las bifurcaciones posibles, marcaban el camino oculto que llevaba de los axiomas a los teoremas.

Forma típica de un teorema: Si se cumple un conjunto de hipótesis, entonces se verifica una tesis.

Corolario: Conclusión (más o menos) inmediata de un teorema

Lema: Sentencia intermedia de la demostración (que pudo haber sido un teorema por demostrar)

Un hermoso ejemplo

Un sistema formal consiste de los símbolos {M, I, U} y algunas reglas gramaticales para construir cadenas válidas a partir de otras conocidas con anterioridad:

Axioma : MI

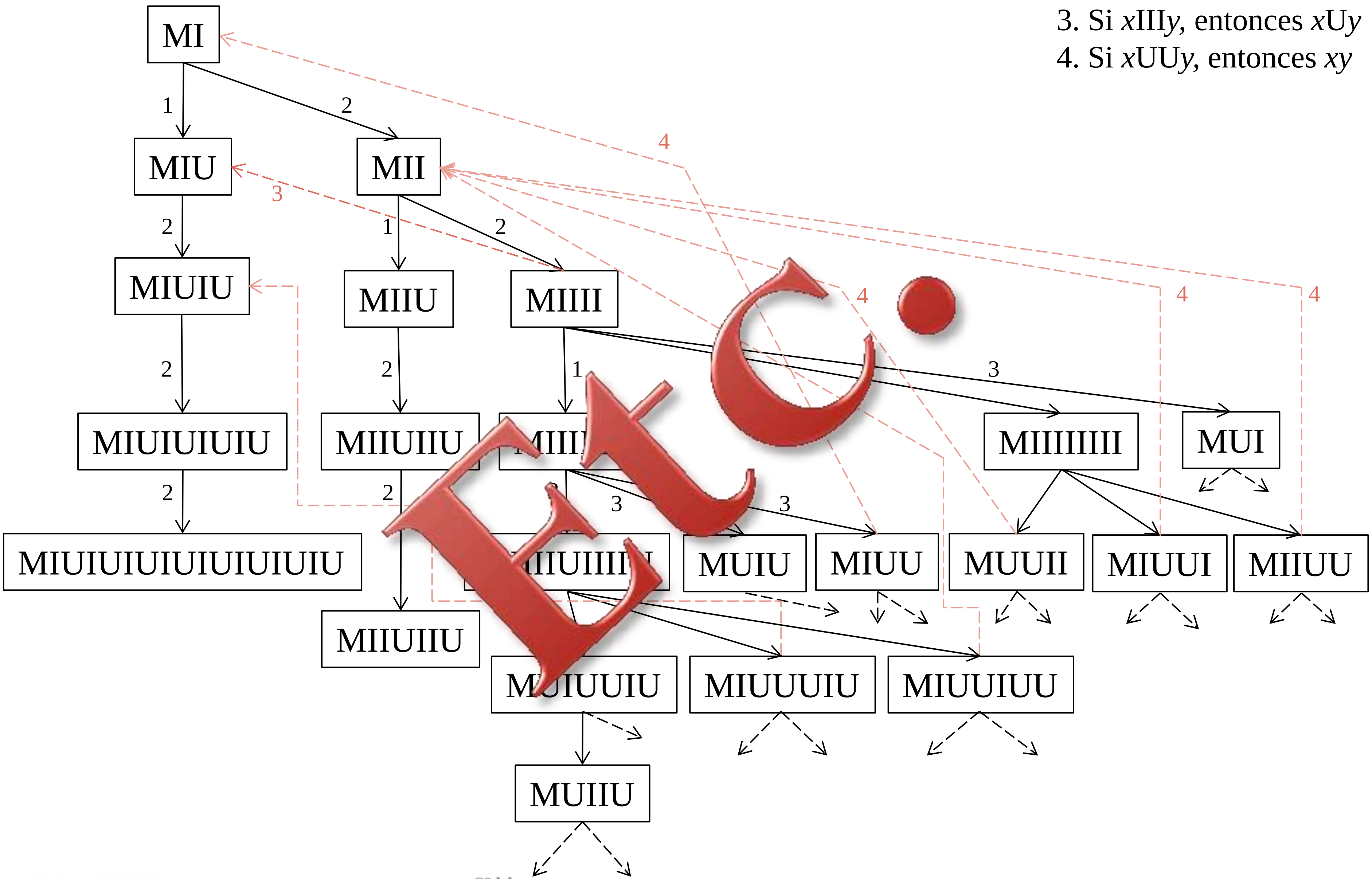
Reglas gramaticales :
1. Si *x*I, entonces *x*IU
2. Si M*x*, entonces M*xx*
3. Si *x*III*y*, entonces *x*U*y*
4. Si *x*UU*y*, entonces *xy*

Tarea : Demuestre MU, si es posible

(donde *x* & *y* son cualquier secuencia –aún secuencias vacías– de símbolos M, I y/o U)

Douglas R. Hofstadter,
"Gödel, Escher, Bach: an Eternel Golden Braid",
Basic Books, 1979

1. Si xI, entonces xIU
2. Si Mx, entonces Mxx
3. Si xIIIy, entonces xUy
4. Si xUUy, entonces xy



Douglas R. Hofstadter,
"Gödel, Escher, Bach: an Eternel Golden Braid",
Basic Books, 1979

3 maneras: **M**echanical mode
I ntelligent mode
U n-mode

El modo **M**ecánico aplica las reglas hasta que aparezca MU… ¡o por toda la eternidad si MU no aparece!

El modo **I**nteligente empieza observando patrones mientras se gana experiencia con el sistema. Por ejemplo, ninguna regla exige que los teoremas empiecen con **M**, pero la relación entre el axioma inicial y las reglas de producción hacen obvio que los teoremas deben empezar con **M**.

Es imposible llegar a la cadena **MU** desde la cadena **MI** mediante la aplicación repetida de las reglas dadas. Para probar esta afirmación basta con mirar el número de **I**es en una cadena:

- Sólo las reglas 2 y 3 cambian este número.
- La regla 2 lo dobla, mientras que le regla 3 lo reduce en 3.
- Entonces hay una propiedad de invarianza: El número de **I**es no es divisible por 3:
  - Al comienxo, el número de **I**es es 1, que no es divisible por 3.
  - Doblar un número que no es divisible por 3 no lo hace divisible por 3.
  - Restar 3 de un número que no es divisible por 3 tampoco lo hace divisible por 3.
  - Entonces, el objetivo **MU**, con cero **I**es, no se puede conseguir porque 0 es divisible por 3.

El axioma **MI** tiene una **I**

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

La regla 2 duplica el número de **I**es

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

(32)

La regla 3 elimina 3*n* **I**es

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

(20,23,26,29,32)

La regla 2 duplica el número de **I**es

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

(20,22,23,26,28,29,32,34)

La regla 3 elimina 3*n* **I**es

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

(20,22,23,25,26,28,29,31,32,34)

el número de **I**es no puede ser divisible por 3

Douglas R. Hofstadter,
“Gödel, Escher, Bach: an Eternel Golden Braid”,
Basic Books, 1979

3 maneras: **M**echanical mode
I ntelligent mode
U n-mode

El modo **M**ecánico aplica las reglas hasta que aparezca MU… ¡o por toda la eternidad si MU no aparece!

El modo **I**nteligente aplica un meta-formalismo para demostrar que MU no es posible en el sistema MIU.

¿Y el **U**n-mode? ¡El modo del Zen!

El término Japonés y Koreano ***mu*** (Japones: 無; Koreano: 무) o Chino ***wu*** (chino tradicional: 無; chino simplificado: 无) significa “No tener”, “Sin”. Es una palabra clave en el Budismo, especialmente en la tradición Zen

Algunas traducciones: “no”, “nada”, “sin”, “No existencia”, “no ser”, “no tener”, “falta de”, “imposible”, “sin razón ni causa”, “pura conciencia humana”, “anterior a la experiencia del conocimiento”. El Zen dice que la palabra **MU** es “la puerta a la iluminación". Por eso el koan **MU** se considera hosshin 発心 "*resuélvase para obtener la iluminación*", esto es, es apropiado para principiantes en busca de kenshō (en busca de la “naturaleza Buda”).

Un colección de koanes del siglo 13 usa la palabra **MU** en su título (Mumonkan 無門關). El primer koan es “el perro de Joshu” 趙州狗子.

Un monje pregunto al maestro Zen *Jōshū* “¿Tiene un perro naturaleza Buda?” *Jōshū* repondió *“Mu"*

Douglas R. Hofstadter,
"Gödel, Escher, Bach: an Eternel Golden Braid",
Basic Books, 1979

3 maneras: **M**echanical mode
I ntelligent mode
U n-mode

El modo **M**ecánico aplica las reglas hasta que aparezca MU… ¡o por toda la eternidad si MU no aparece!

El modo **I**nteligente aplica un meta-formalismo para demostrar que MU no es posible en el sistema MIU.

El **U**n-modo permite que la demostración emerja como la "iluminación" que proviene de la "nada".

Sistemas formales como números

Un sistema formal consiste de los símbolos {M, I, U} y algunas reglas gramaticales para construir cadenas válidas a partir de otras conocidas con anterioridad:

Axioma : MI
Reglas gramaticales : 1. Si xI, entonces xIU
2. Si Mx, entonces Mxx
3. Si xIIIy, entonces xUy
4. Si xUUy, entonces xy
Tarea : Demuestre MU, si es posible

(donde x & y son cualquier secuencia –aún secuencias vacías– de símbolos M, I y/o U)

Un sistema formal consiste de los símbolos {3, 1, 0} y algunas reglas gramaticales para construir cadenas válidas a partir de otras conocidas con anterioridad:

(donde $k, m, n \in \mathrm{N}$ y $n < 10^m$)

Axioma : 31
Reglas gramaticales : 1. Si $10\cdot k+1$, entonces $10\cdot(10\cdot k+1)$
2. Si $3\cdot 10^m+n$, entonces $10^m\cdot(3\cdot 10^m+n) + n$
3. Si $k\cdot 10^{m+3}+111\cdot 10^m+n$, entonces $k\cdot 10^{m+1}+n$
4. Si $k\cdot 10^{m+2}+n$, entonces $k\cdot 10^m+n$
Tarea : Demuestre 30, si es posible

Algunos sistemas formales como estos usados en el modelado matemático de sistemas complejos

Sistemas-*L*: Gramática Formal y Re-escritura

Chomsky
Gramática formal
Re-escribir para describir las características sintácticas del lenguaje natural (1950's)

Backus y Naur
Notación basada en re-escritura
Definición formal de los lenguajes de programación (Algol, 1960's)

El reconocimiento de su equivalencia inició un período de fascinación con la sintaxis, la gramática y su aplicación a la ciencia de la computación

Sistemas-L: Re-escritura

- Un árbol crece desde una semilla: ¿Cómo se pueden generar nuevas células a partir de las células viejas?
- La semilla se conoce como un *axioma*
- Las instrucciones para hacer crecer nuevas células se conocen como *reglas de producción*
- Por ejemplo,

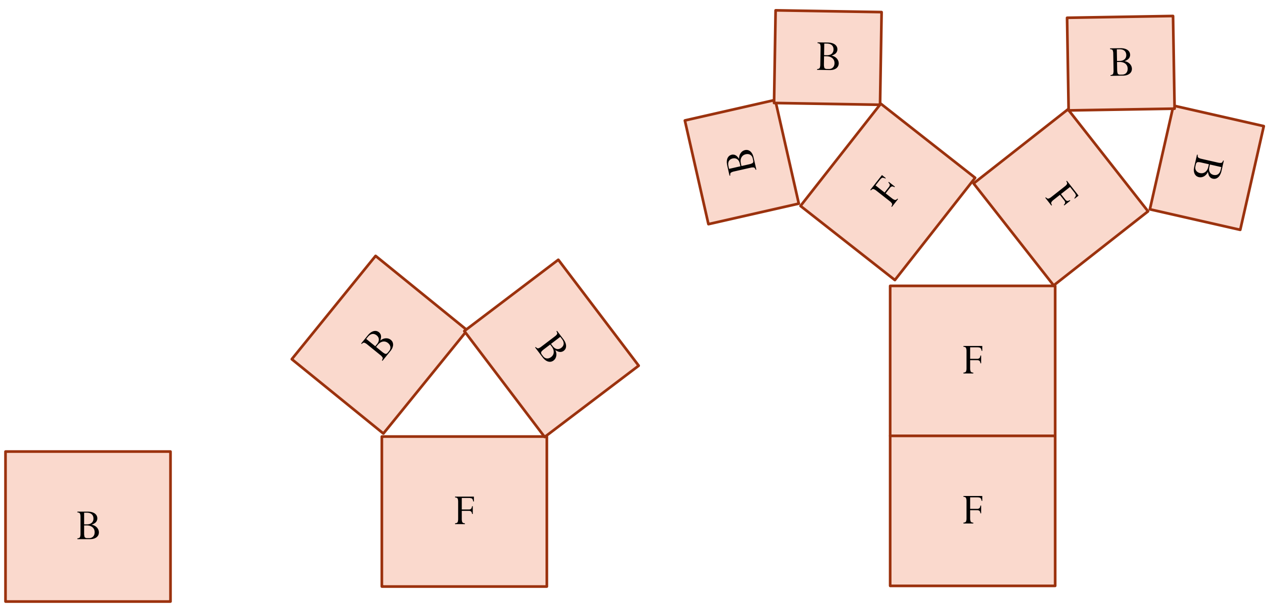
Axioma : B

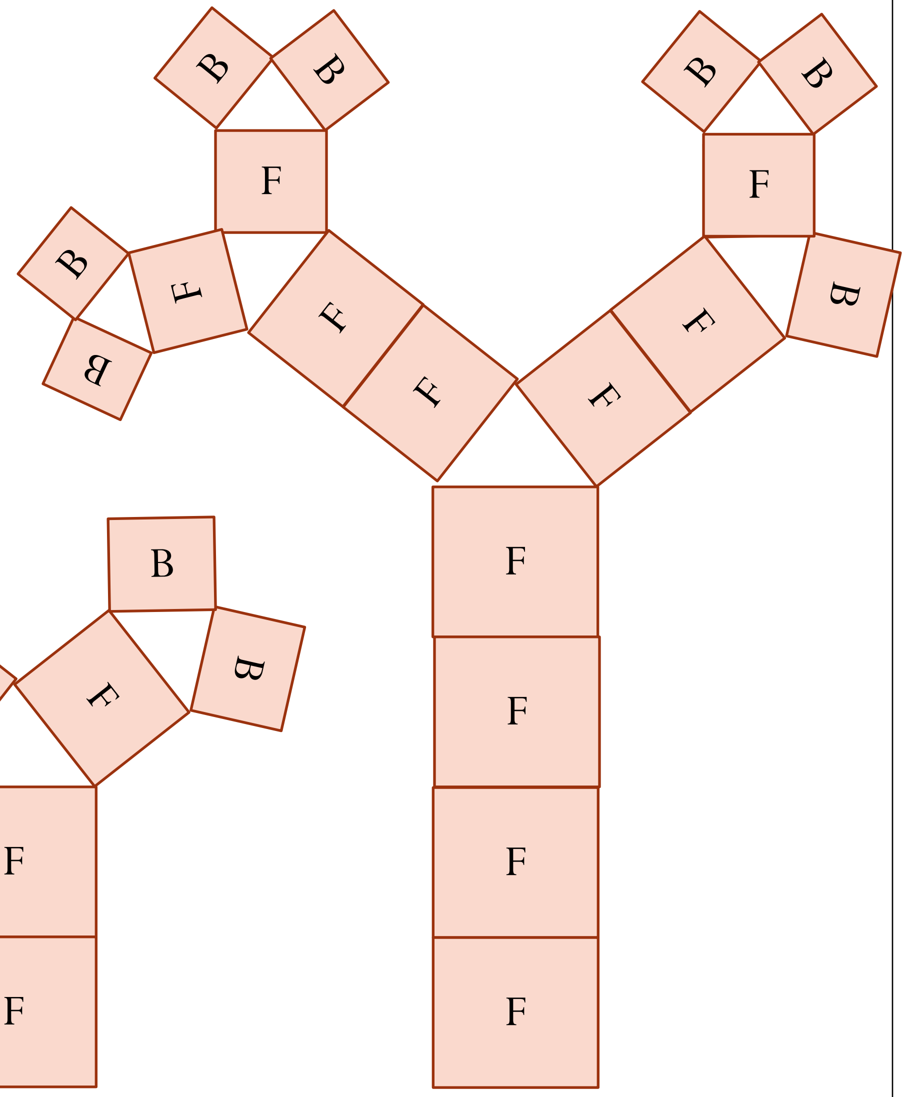
Reglas : B→F-B+B, F→FF

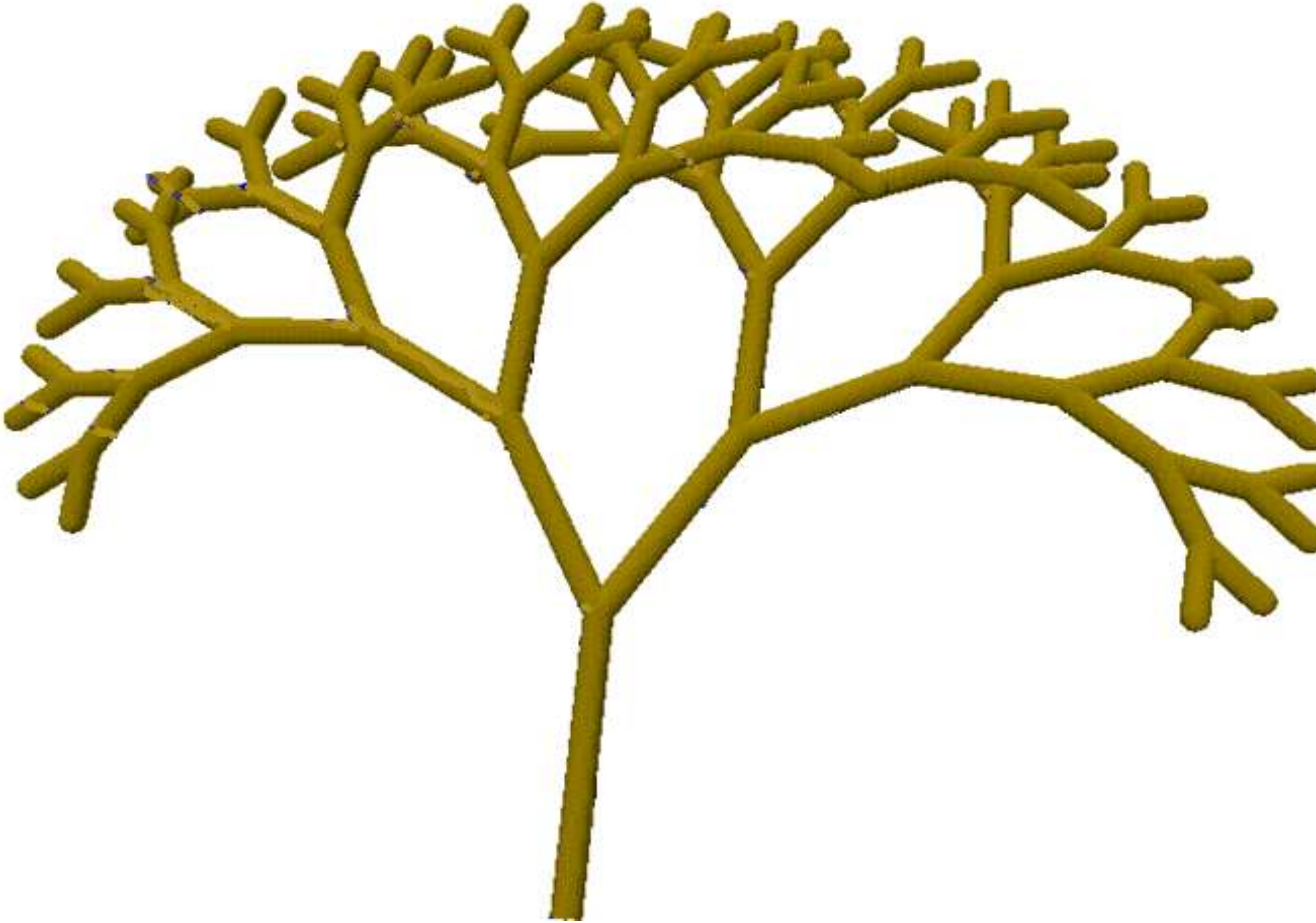
Nivel	Cadena
0	B
1	F-B+B
2	FF-(F-B+B)+(F-B+B)
3	FFFF-(FF-(F-B+B)+(F-B+B))+(FF-(F-B+B)+(F-B+B))

Sistemas-L: Re-escritura

Nivel	Cadena
0	B
1	F-B+B
2	FF-(F-B+B)+(F-B+B)
3	FFFF-(FF-(F-B+B)+(F-B+B))+(FF-(F-B+B)+(F-B+B))







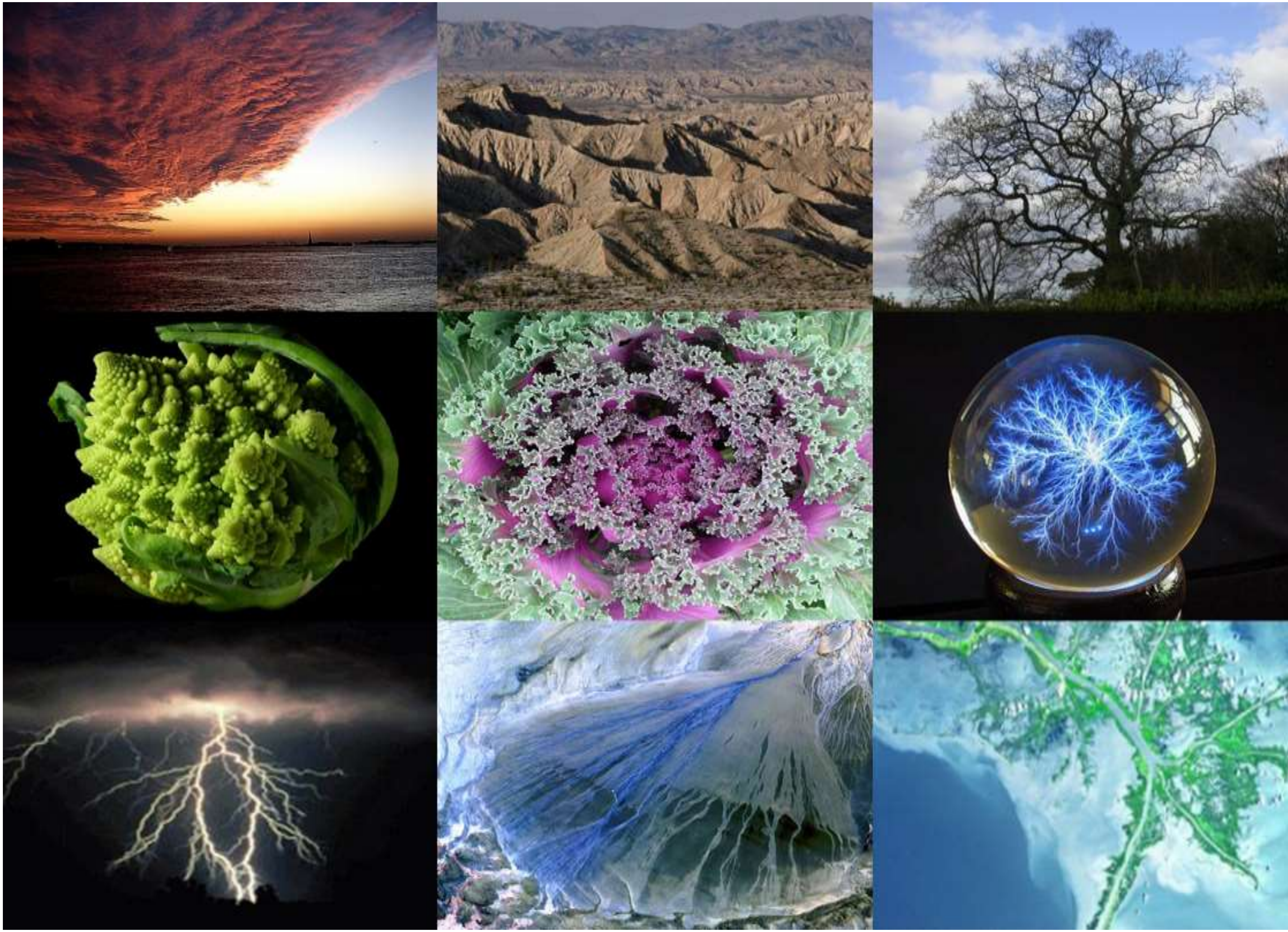
Sistemas L

Aristid Lindenmayer, 1925 - 1989

THE ALGORITHMIC BEAUTY OF PLANTS

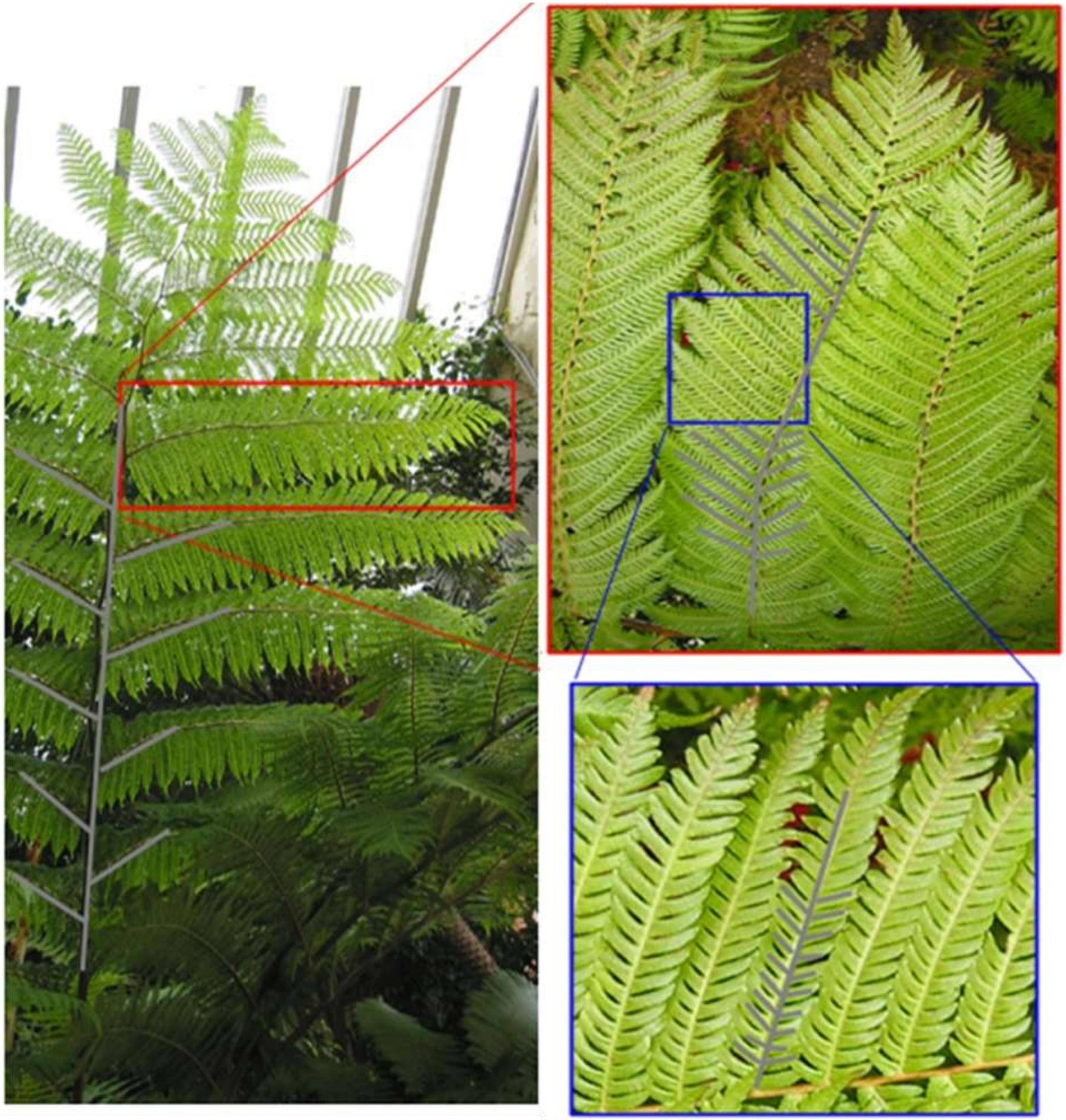
PRZEMYSLAW PRUSINKIEWICZ • ARISTID LINDENMAYER

En muchos procesos de crecimiento de organismos vivos, especialmente de plantas, se puede notar fácilmente la aparición de ciertas estructuras multicelulares que se repiten regularmante.... En el caso de una hoja compuesta, por ejemplo, algunos lóbulos, que son partes de la hoja en un estado avanzado, tienen la misma forma que la hoja entera tenía en una etapa anterior… Matemáticamente hablando, la forma orgánica de la hoja es una función del tiempo... Podemos decir que la forma de un organismo es un evento en el espacio-tiempo y no solamente una configuración en el espacio… *La idea de una forma contiene implícitamente la historia de la forma*.



¿Otra geometría?

- **Las formas principales no se construyen como líneas rectas, círculos, triángulos, elipses, etc.**
- **… Se parecen más a un conjunto de procedimientos (algoritmos) para rotar, desplazar, re-escalizar o distorsionar alguna forma original.**



Transformaciones afines en $\mathbb{R}^2$

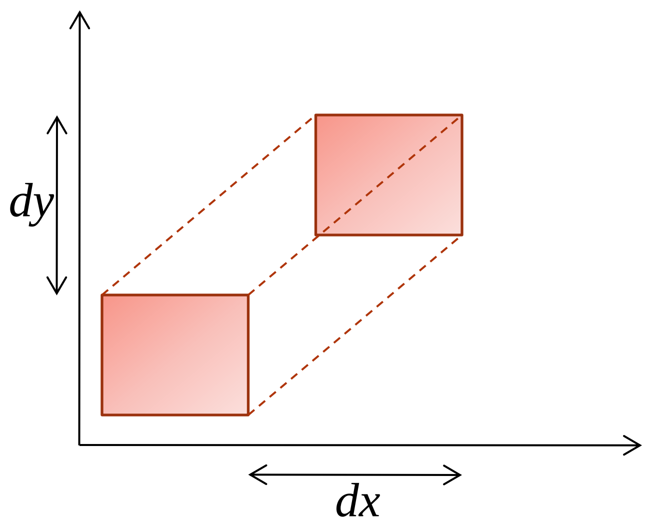
$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix}$$

$\downarrow$

Rotar
Escalizar
distorsionar

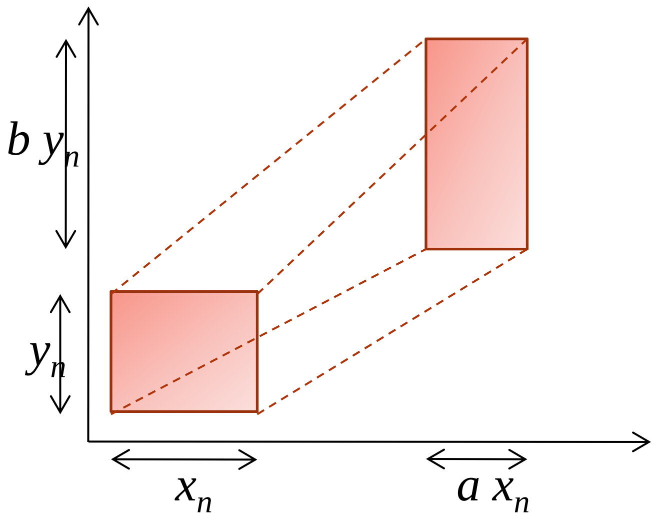
$\downarrow$

desplazar



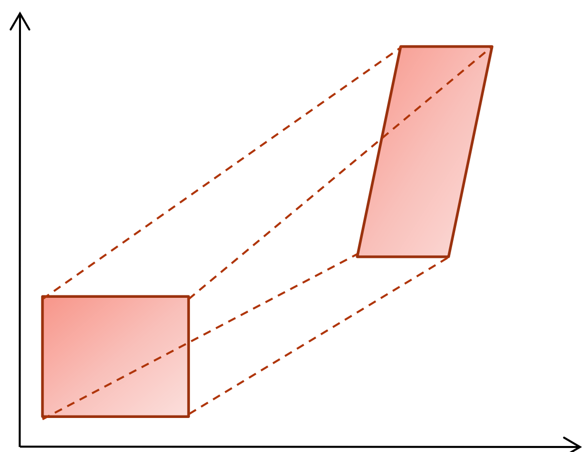
$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} + \begin{bmatrix} dx \\ dy \end{bmatrix}$$

Una translación de (dx,dy)



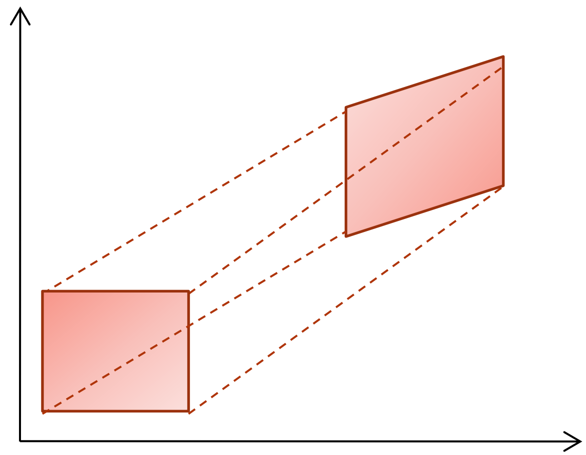
$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} + \begin{bmatrix} dx \\ dy \end{bmatrix}$$

Re-escalización por (a,b)



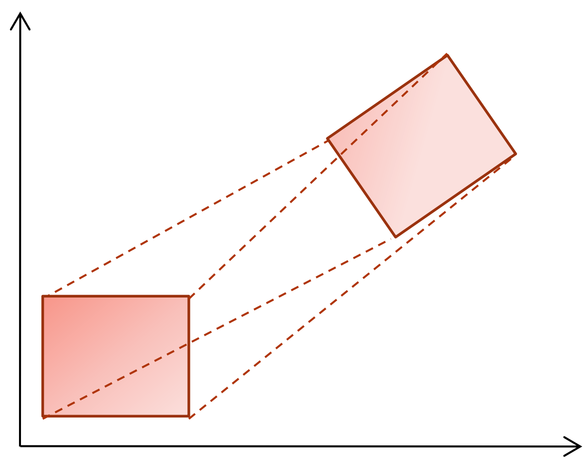
$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} a & b \\ 0 & c \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} + \begin{bmatrix} dx \\ dy \end{bmatrix}$$

Deformación paralela a x



$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} a & 0 \\ b & c \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} + \begin{bmatrix} dx \\ dy \end{bmatrix}$$

Deformación paralela a *y*



$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} + \begin{bmatrix} dx \\ dy \end{bmatrix}$$

Rotación por un ángulo θ

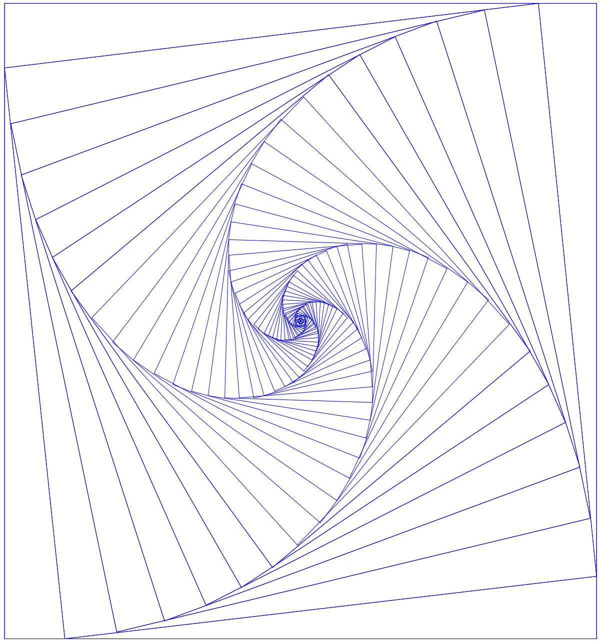
Iteración de una transformación simple

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 0.9 & -0.1 \\ 0.1 & 0.9 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} + \begin{bmatrix} 0.1 \\ 0.0 \end{bmatrix}$$

Iterated

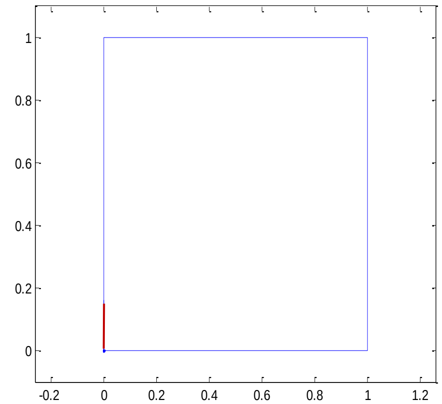
Function

System

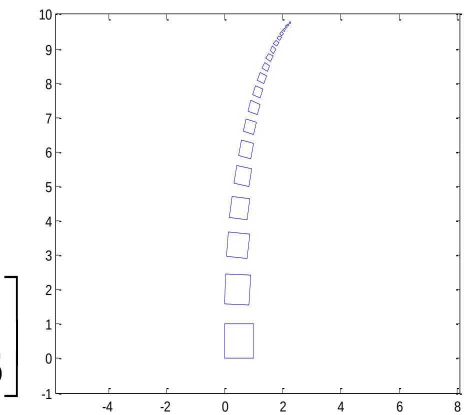


```
xy     = [0 0; 1 0; 1 1; 0 1; 0 0]';
A      = [0.9 -0.1; 0.1 0.9];
b      = repmat([0.1; 0.0],1,5);
for i = 1:100
    plot(xy(1,:),xy(2,:)); hold on
    xy = A*xy + b;
end
```

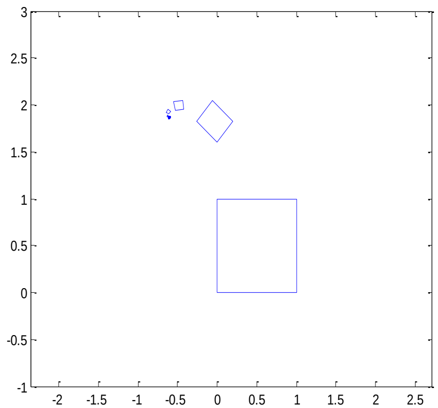
Cuatro transformaciones "interesantes" (¿?)



$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0.16 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

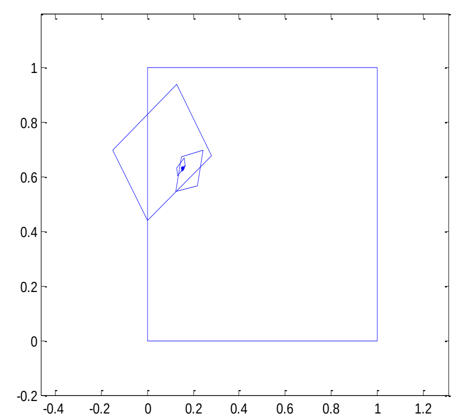


$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 0.85 & 0.04 \\ -0.04 & 0.85 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} + \begin{bmatrix} 0 \\ 1.6 \end{bmatrix}$$



$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 0.2 & -0.26 \\ 0.23 & 0.22 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} + \begin{bmatrix} 0 \\ 1.6 \end{bmatrix}$$

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} -0.15 & 0.28 \\ 0.26 & 0.24 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} + \begin{bmatrix} 0 \\ 0.44 \end{bmatrix}$$



Geometría de la naturaleza

Empezando con **x** = (0.5, 0.5):

  grafique(**x**)

  con prob. 0.01, $\mathbf{x} \leftarrow A_1\,\mathbf{x} + b_1$;

  con prob. 0.85, $\mathbf{x} \leftarrow A_2\,\mathbf{x} + b_2$;

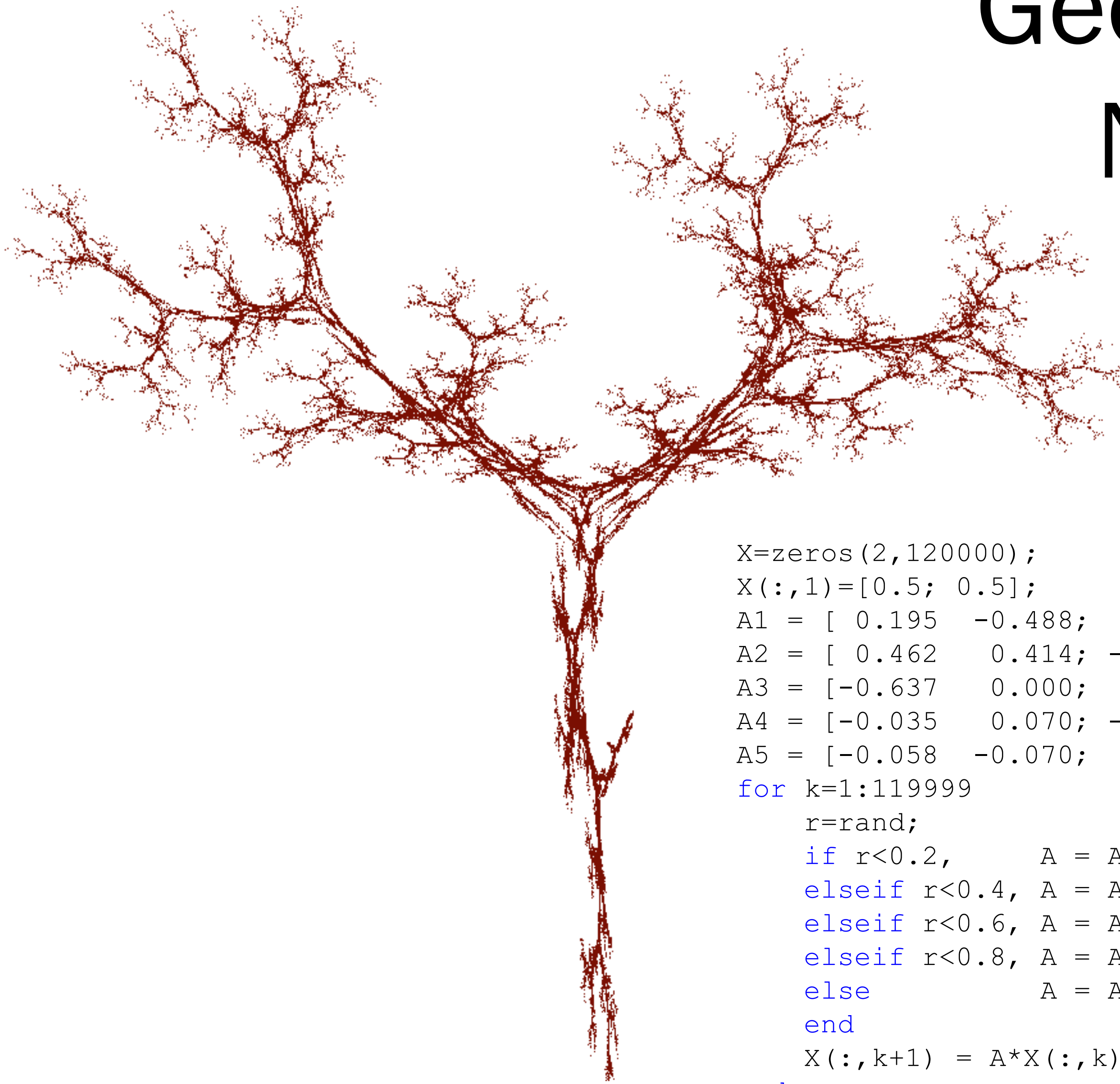
  con prob. 0.07, $\mathbf{x} \leftarrow A_3\,\mathbf{x} + b_3$;

  con prob. 0.07, $\mathbf{x} \leftarrow A_4\,\mathbf{x} + b_4$;

Repita hasta tener suficientes puntos

```
X=zeros(2,120000);
X(:,1)=[0.5; 0.5];
A1 = [ 0.00  0.00;  0.00  0.16]; b1 = [0.0; 0.00];
A2 = [ 0.85  0.04; -0.04  0.85]; b2 = [0.0; 1.60];
A3 = [ 0.20 -0.26;  0.23  0.22]; b3 = [0.0; 1.60];
A4 = [-0.15  0.28;  0.26  0.24]; b4 = [0.0; 0.44];
for k=1:119999
    r=rand;
    if     r<.01, A = A1; b = b1;
    elseif r<.86, A = A2; b = b2;
    elseif r<.93, A = A3; b = b3;
    else          A = A4; b = b4;
    end
    X(:,k+1) = A*X(:,k) + b;
end
scatter(X(1,:),X(2,:),2,'g')
axis equal
```

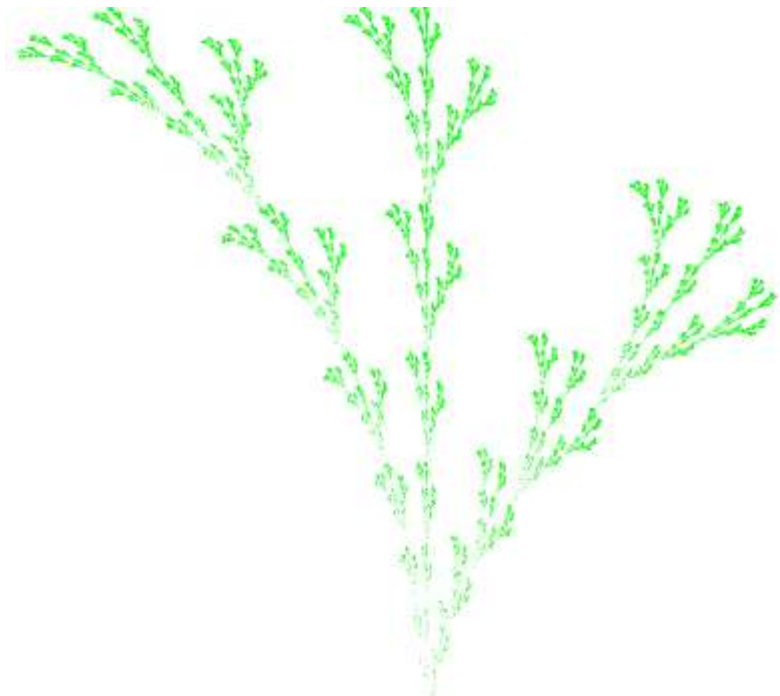
Geometría de la Naturaleza



```
X=zeros(2,120000);
X(:,1)=[0.5; 0.5];
A1 = [ 0.195  -0.488;  0.344  0.443]; b1 = [0.4431; 0.2452];
A2 = [ 0.462   0.414; -0.252  0.361]; b2 = [0.2511; 0.5692];
A3 = [-0.637   0.000;  0.000  0.501]; b3 = [0.8562; 0.2512];
A4 = [-0.035   0.070; -0.469  0.022]; b4 = [0.4884; 0.5069];
A5 = [-0.058  -0.070;  0.453 -0.111]; b5 = [0.5976; 0.0969];
for k=1:119999
    r=rand;
    if r<0.2,      A = A1; b = b1;
    elseif r<0.4,  A = A2; b = b2;
    elseif r<0.6,  A = A3; b = b3;
    elseif r<0.8,  A = A3; b = b3;
    else           A = A5; b = b5;
    end
    X(:,k+1) = A*X(:,k) + b;
end
scatter(X(1,:),X(2,:),2,'r')
axis equal
```

```
X=zeros(2,120000);
X(:,1)=[0.5; 0.5];
A1 = [ 0.387  0.430;  0.430 -0.387]; b1 = [0.2560; 0.5220];
A2 = [ 0.441 -0.091; -0.009 -0.322]; b2 = [0.4219; 0.5059];
A3 = [-0.468  0.020; -0.113  0.015]; b3 = [0.4000; 0.4000];
for k=1:119999
    r=rand;
    if r<.333,      A = A1; b = b1;
    elseif r<.666, A = A2; b = b2;
    else            A = A3; b = b3;
    end
    X(:,k+1) = A*X(:,k) + b;
end
scatter(X(1,:),X(2,:),2,'g')
```

```
X=zeros(2,120000);
X(:,1)=[0.5; 0.5];
A1 = [ 0.50  0.00;  0.00  0.75]; b1 = [0.25; 0.00];
A2 = [ 0.25 -0.20;  0.10  0.30]; b2 = [0.25; 0.50];
A3 = [ 0.25  0.20; -0.10  0.30]; b3 = [0.50; 0.40];
A4 = [ 0.20  0.00;  0.00  0.30]; b4 = [0.40; 0.55];
for k=1:119999
    r=rand;
    if r<0.25,      A = A1; b = b1;
    elseif r<0.50, A = A2; b = b2;
    elseif r<0.75, A = A3; b = b3;
    else            A = A4; b = b4;
    end
    X(:,k+1) = A*X(:,k) + b;
end
scatter(X(1,:),X(2,:),2,'g')
axis equal
```



Los teoremas de Gödel no nos quitaron el uso de sistemas formales para el modelado matemático de sistemas complejos

¡¡Al Contrario!!

Modo **M**ecánico

Modo **I**nteligente

Modo **U**n-mode

¡Los teoremas de Gödel nos permiten descubrir nuevas y maravillosas maneras de trabajar con los sistemas formales que representan sistemas complejos!

Esto es todo respecto a sistemas formales

$A_1, A_2, \ldots, A_k$ - axiomas y teoremas demostrados previamente

La demostración formal de un enunciado P es una secuencia de enunciados

$$S_1, S_2, \ldots, S_n$$

donde:

1. S_n es P y ocurre uno de las siguientes dos hechos:

2a. S_i es alguno de los axiomas $A_1, A_2, \ldots, A_k$

o

2b. S_i se deduce de los enunciados anteriores mediante una argumentación válida usando las reglas de razonamiento (reglas gramaticales)

La matemática es un lenguaje (formal)

Ejemplo: Sea $(\Omega, \mathscr{F}, \mathbf{P})$ un espacio de probabilidad

Axiomas: (1) $\mathbf{P}(\Omega) = 1$

(2) $(A \in \mathscr{F}) \Rightarrow \mathbf{P}(A) \geq 0$

(3) $((A, B \in \mathscr{F}) \wedge (A \cap B = \phi)) \Rightarrow \mathbf{P}(A \cup B) = \mathbf{P}(A) + \mathbf{P}(B).$

Teorema: $(A \subseteq B) \Rightarrow \mathbf{P}(A) \leq \mathbf{P}(B).$

Demostración: $B = A \cup (B \cap A^C)$

$\mathbf{P}(B) = \mathbf{P}(A) + \mathbf{P}(B \cap A^C)$

$\mathbf{P}(B \cap A^C) \geq 0$

$\mathbf{P}(B) \geq \mathbf{P}(A).$ **QED □**

Una página cualquiera (la 203, p.ej.) de *Principia Mathematica*

$*20{\cdot}23.\quad \vdash : \hat{z}(\phi z) = \hat{z}(\psi z) \,.\, \hat{z}(\phi z) = \hat{z}(\chi z) \,.\supset .\, \hat{z}(\psi z) = \hat{z}(\chi z) \qquad [*20{\cdot}21{\cdot}22]$

$*20{\cdot}24.\quad \vdash : \hat{z}(\psi z) = \hat{z}(\phi z) \,.\, \hat{z}(\chi z) = \hat{z}(\phi z) \,.\supset .\, \hat{z}(\psi z) = \hat{z}(\chi z) \qquad [*20{\cdot}21{\cdot}22]$

$*20{\cdot}25.\quad \vdash :. \alpha = \hat{z}(\phi z) \,.\equiv_\alpha .\, \alpha = \hat{z}(\psi z) : \equiv .\, \hat{z}(\phi z) = \hat{z}(\psi z)$

Dem.

$\vdash . *10{\cdot}1 . \quad \supset \vdash :. \alpha = \hat{z}(\phi z) \,.\equiv_\alpha .\, \alpha = \hat{z}(\psi z) : \supset :$

$\hat{z}(\phi z) = \hat{z}(\phi z) \,.\equiv .\, \hat{z}(\phi z) = \hat{z}(\psi z) :$

$[*20{\cdot}2] \quad \supset : \hat{z}(\phi z) = \hat{z}(\psi z) \qquad (1)$

$\vdash . *20{\cdot}22 . \quad \supset \vdash : \alpha = \hat{z}(\phi z) \,.\, \hat{z}(\phi z) = \hat{z}(\psi z) \,.\supset .\, \alpha = \hat{z}(\psi z) :$

$[\text{Exp.Comm}] \supset \vdash :. \hat{z}(\phi z) = \hat{z}(\psi z) \,.\supset : \alpha = \hat{z}(\phi z) \,.\supset .\, \alpha = \hat{z}(\psi z) \qquad (2)$

$\vdash . *20{\cdot}24 . \quad \supset \vdash :. \hat{z}(\phi z) = \hat{z}(\psi z) \,.\, \alpha = \hat{z}(\psi z) \,.\supset .\, \alpha = \hat{z}(\phi z) :.$

$[\text{Exp}] \quad \supset \vdash :. \hat{z}(\phi z) = \hat{z}(\psi z) \,.\supset : \alpha = \hat{z}(\psi z) \,.\supset .\, \alpha = \hat{z}(\phi z) \qquad (3)$

$\vdash . (2) . (3) . \quad \supset \vdash :. \hat{z}(\phi z) = \hat{z}(\psi z) \,.\supset : \alpha = \hat{z}(\phi z) \,.\equiv .\, \alpha = \hat{z}(\psi z) :.$

$[*10{\cdot}11{\cdot}21] \quad \supset \vdash :. \hat{z}(\phi z) = \hat{z}(\psi z) \,.\supset : \alpha = \hat{z}(\phi z) \,.\equiv_\alpha .\, \alpha = \hat{z}(\psi z) \qquad (4)$

$\vdash . (1) . (4) . \quad \supset \vdash . \boxed{\text{Prop}}$

$*20{\cdot}3.\quad \vdash : x \in \hat{z}(\psi z) \,.\equiv .\, \psi x$

Dem.

$\vdash . *20{\cdot}1 . \supset$

$\vdash :: x \in \hat{z}(\psi z) \,.\equiv :. (\exists \phi) :. \psi y \,.\equiv_y .\, \phi ! y : x \in (\phi ! \hat{z}) :.$

$[(*20{\cdot}02)] \quad \equiv :. (\exists \phi) :. \psi y \,.\equiv_y .\, \phi ! y : \phi ! x :.$

$[*10{\cdot}43] \quad \equiv :. (\exists \phi) :. \psi y \,.\equiv_y .\, \phi ! y : \psi x :.$

$[*10{\cdot}35] \quad \equiv :. (\exists \phi) : \psi y \,.\equiv_y .\, \phi ! y :. \psi x :.$

$[*12{\cdot}1] \quad \equiv :. \psi x :: \supset \vdash . \boxed{\text{Prop}}$

$*20{\cdot}31.\quad \vdash :. \hat{z}(\psi z) = \hat{z}(\chi z) \,.\equiv : x \in \hat{z}(\psi z) \,.\equiv_x .\, x \in \hat{z}(\chi z) \qquad [*20{\cdot}15{\cdot}3]$

$*20{\cdot}32.\quad \vdash . \hat{x}\{x \in \hat{z}(\phi z)\} = \hat{z}(\phi z) \qquad [*20{\cdot}3{\cdot}15]$

$*20{\cdot}33.\quad \vdash :. \alpha = \hat{z}(\phi z) \,.\equiv : x \in \alpha \,.\equiv_x .\, \phi x$

Dem.

$\vdash . *20{\cdot}31 . \quad \supset \vdash :. \alpha = \hat{z}(\phi z) \,.\equiv : x \in \alpha \,.\equiv_x .\, x \in \hat{z}(\phi z) \qquad (1)$

$\vdash . (1) . *20{\cdot}3 . \supset \vdash . \boxed{\text{Prop}}$

$*20{\cdot}34.\quad \vdash :. x = y \,.\equiv : x \in \alpha \,.\supset_\alpha .\, y \in \alpha$

Dem.

$\vdash . *4{\cdot}2 . (*20{\cdot}07) . \supset \vdash :. x \in \alpha \,.\supset_\alpha .\, y \in \alpha : \equiv : x \in \hat{z}(\phi ! z) \,.\supset_\phi .\, y \in \hat{z}(\phi ! z) :$

$[*20{\cdot}3] \quad \equiv : \phi ! x \,.\supset_\phi .\, \phi ! y :$

$[*13{\cdot}1] \quad \equiv : x = y :. \supset \vdash . \boxed{\text{Prop}}$

QED □

Matemáticas inductivas

Mesopotamia, 2000 AC

Lista de valores enteros *a,b,c* que satisfacen

$$a^2+b^2=c^2$$

con instrucciones para realizar excavaciones de sistemas de riego.
No hay demostraciones, sólo hechos.

Egipcios
Mayas

Matemáticas deductivas

Pitágoras, Euclides, Thales…

Teorema de Thales – Les Luthiers

Si tres o más paralelas
Si tres o más parale-le-le-las
Si tres o más paralelas
Si tres o más parale-le-le-las
son cortadas, son cortadas
por dos transversales, dos transversales,
son cortadas, son cortadas
por dos transversales, dos transversales,
Si tres o más parale-le-le-las
Si tres o más parale-le-le-las
son cortadas, son cortadas,
son cortadas, son cortadas
Dos segmentos de una de estas, dos segmentos cualesquiera,
dos segmentos de una de estas son proporcionales
a los dos segmentos correspondientes de la otra.

Hipótesis:

A paralela a B, B paralela a C,
A paralela a B, paralela a C, paralela a D.
O-P es a P-Q, M-N es a N-T,
OP es a PQ como MN es a NT.

A paralela a B,
B paralela a C,
OP es a PQ como MN es a NT.
La bisectriz yo trazaré
y a cuatro planos intersectaré.
Una igualdad yo encontraré:
OP+PQ es igual a ST.
Usaré la hipotenusa.
Ay, no te compliques, nadie la usa.
Trazaré, pues, un cateto.
Yo no me meto, yo no me meto.

Triángulo, tetrágono, pentágono, hexágono,
heptágono, octógono, son todos polígonos.
Seno, coseno, tangente y secante,
y la cosecante y la cotangente.

Thales, Thales de Mileto
Thales, Thales de Mileto
Thales, Thales de Mileto
Thales, Thales de Mileto

QED □

Que es lo que queríamos demostrar.
Queesque loqueloque queriariamos demodemostrar!

Una características fundamental

- **El conjunto de axiomas debe ser recursivo**
  - En una cantidad finita de pasos se puede determinar si un enunciado cualquiera pertenece o no al conjunto de axiomas
  - La recursividad es fundamental porque
    - Toda demostración a partir de un conjunto recursivo de axiomas podría corroborarse en un número finito de pasos
    - Todas las demostraciones a que puede dar lugar un conjunto recursivo de axiomas podrían ser generadas mecánicamente por una computadora
  - Por ejemplo, un conjunto finito de axiomas es recursivo
    - Compare el enunciado dado con cada uno de los axiomas
  - Por ejemplo, un conjunto infinito contable de axiomas en el que la longitud de los axiomas es creciente, es recursivo
    - Compare el enunciado con los axiomas de su misma longitud

Una característica deseable y otra necesaria

- Un conjunto de objetos matemáticos, $X = \{\ldots\}$
- Un conjunto de enunciados verdaderos denominados axiomas $A(X)$
- Un conjunto de reglas de producción

Un sistema formal

Deseable

- El sistema es completo si todos los enunciados verdaderos sobre el conjunto X se pueden demostrar como teoremas
  - El sistema es completo si todos los enunciados son decidibles (si se puede decidir en un número finito de pasos si es un teorema o no)

Necesaria

- El sistema es consistente si no da lugar a contradicciones: Si permite demostrar $T(X)$, no debe permitir demostrar $\sim T(X)$

Douglas R. Hofstadter,
"Gödel, Escher, Bach: an Eternel Golden Braid",
Basic Books, 1979

El sistema *pq*:

- Tres símbolos distintos
  p q –
- Un número infinito de axiomas:
  *x*p–q*x*– es un axioma siempre que *x* esté compuesto sólo de rayas (1 o más)
- Una regla única
  Suponga que *x*, *y* & *z* son cadenas particulares de rayas. Si se sabe que *x*p*y*q*z* es un teorema, entonces *x*p*y*–q*z*– también es un teorema

Ejemplos:

Como –p–q–– es un axioma ⇒ –p––q––– es un teorema

Como –––p–q–––– es un axioma ⇒ –––p––q––––– es un teorema

Si ––p––––q–––––– es un teorema ⇒ ––p–––––q––––––– también lo es

Pero ¿cómo sabemos si ––p––––q–––––– es un teorema?

––p–q––– ⇒ ––p––q–––– ⇒ ––p–––q––––– ⇒ ––p––––q––––––

¡Recursivo, completo y consistente!

ideas GITUD

Gödel: Metáforas y realidades

Marco Aurelio Alzate Monroy

¿Imaginan un procedimiento **I**nteligente para distinguir entre teoremas y no-teoremas?

Douglas R. Hofstadter,
"Gödel, Escher, Bach: an Eternel Golden Braid",
Basic Books, 1979

¿Imaginan un procedimiento Inteligente para distinguir entre teoremas y no-teoremas?

Los primeros dos grupos de rayas deben igualar en longitud al tercer grupo de rayas

- Los axiomas satisfacen el criterio de sumar 1
- En la regla gramatical, si el primer enunciado satisface el criterio de la suma, el segundo enunciado también lo hace
- De otro lado, si el primer enunciado no satisface el criterio de la suma, el segundo enunciado tampoco lo hace

El isomorfismo induce el significado

```
    p ⇔ +
    q ⇔ =
    - ⇔ 1
   -- ⇔ 2
  --- ⇔ 3
   etc.
```

```
---p--q-----
     ⇔
   3+2=5
```

Douglas R. Hofstadter,
"Gödel, Escher, Bach: an Eternel Golden Braid",
Basic Books, 1979

¿Imaginan un procedimiento **I**nteligente para distinguir entre teoremas y no-teoremas?

Los primeros dos grupos de rayas deben igualar en longitud al tercer grupo de rayas

- Los axiomas satisfacen el criterio de sumar 1
- En la regla gramatical, si el primer enunciado satisface el criterio de la suma, el segundo enunciado también lo hace
- De otro lado, si el primer enunciado no satisface el criterio de la suma, el segundo enunciado tampoco lo hace

El isomorfismo induce el significado

p ⇔ +
q ⇔ =
- ⇔ 1
-- ⇔ 2
--- ⇔ 3
etc.

---p--q-----
⇔
3+2=5

p ⇔ es igual a
q ⇔ tomado de
- ⇔ 1
-- ⇔ 2
--- ⇔ 3
etc.

---p--q-----
⇔
3 es igual a 2 tomado de 5

Douglas R. Hofstadter,
“Gödel, Escher, Bach: an Eternel Golden Braid”,
Basic Books, 1979

¿Imaginan un procedimiento **I**nteligente para distinguir entre teoremas y no-teoremas?

Los primeros dos grupos de rayas deben igualar en longitud al tercer grupo de rayas

- Los axiomas satisfacen el criterio de sumar 1
- En la regla gramatical, si el primer enunciado satisface el criterio de la suma, el segundo enunciado también lo hace
- De otro lado, si el primer enunciado no satisface el criterio de la suma, el segundo enunciado tampoco lo hace

El isomorfismo induce el significado

p ⇔ +
q ⇔ =
- ⇔ 1
-- ⇔ 2
--- ⇔ 3
etc.

---p--q-----
⇔
3+2=5

p ⇔ es igual a
q ⇔ tomado de
- ⇔ 1
-- ⇔ 2
--- ⇔ 3
etc.

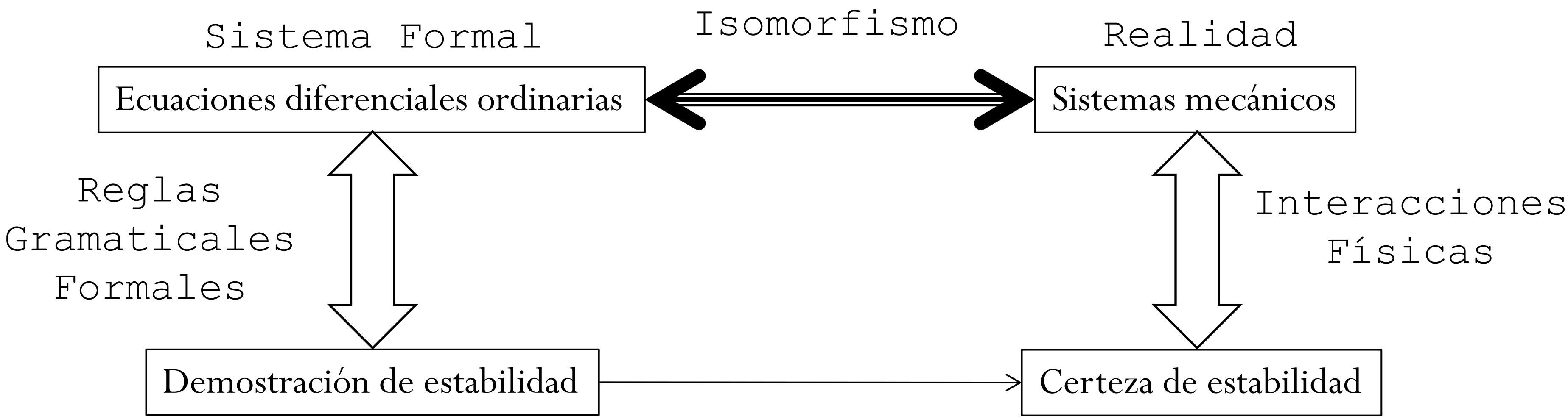
---p--q-----
⇔
3 es igual a 2 tomado de 5

p ⇔ Caballo
q ⇔ Feliz
- ⇔ Manzana

--p-q---
⇔
Manzana Manzana
Caballo
Manzana
Feliz
Manzana Manzana Manzana

¡Sólo si es apropiado!

El isomorfismo induce el significado



Douglas R. Hofstadter,
"Gödel, Escher, Bach: an Eternel Golden Braid",
Basic Books, 1979

El sistema *pq* modificado:

- Tres símbolos distintos

  p q –

- Dos conjuntos infinitos de axiomas:

  *x*p–q*x*– es un axioma siempre que *x* esté compuesto sólo de rayas (1 o más)

  *x*p–q*x* es un axioma siempre que *x* esté compuesto sólo de rayas (1 o más)

- Una regla única

  Suponga que *x*, *y* & *z* son cadenas particulares de rayas. Si se sabe que *x*p*y*q*z* es un teorema, entonces *x*p*y*–q*z*– también es un teorema

Ejemplos:

Como --p-q--- es un axioma ⇒ --p--q---- es un teorema

Como --p-q-- es un axioma ⇒ --p--q--- es un teorema

$2+1=3 \Rightarrow 2+2=4$ ✓

¿2+1=2? ¿2+2=3?

¿Perdimos la consistencia?

Sólo con respecto a la interpretación anterior. Con una nueva interpretación recobra la consistencia

p ⇔ +
q ⇔ ≥
- ⇔ 1
-- ⇔ 2
etc.

--p-q---
⇔
$2+1\geq 3$

--p--q---
⇔
$2+2\geq 3$

Douglas R. Hofstadter,
"Gödel, Escher, Bach: an Eternel Golden Braid",
Basic Books, 1979

El sistema *tq*:

- Tres símbolos distintos
  `t` `q` `-`
- Un número infinito de axiomas:
  x`t-q`*x* es un axioma siempre que *x* esté compuesto sólo de rayas (1 o más)
- Una regla única
  Suponga que *x*, *y* & *z* son cadenas particulares de rayas. Si se sabe que *x*`t`*y*`q`*z* es un teorema, entonces *x*`t`*y*`-q`*zx* también es un teorema

Ejemplos:

Como `-t-q-` es un axioma $\Rightarrow$ `-t--q--` es un teorema

Como `---t-q---` es un axioma $\Rightarrow$ `---t---q---------` es un teorema

En efecto,

`---t-q---` $\Rightarrow$ `---t--q------` $\Rightarrow$ `---t---q---------`

¡Recursivo, completo y consistente!

¿Imaginan un procedimiento **I**nteligente para distinguir entre teoremas y no-teoremas?

Douglas R. Hofstadter,
“Gödel, Escher, Bach: an Eternel Golden Braid”,
Basic Books, 1979

El sistema *tq*:

- Tres símbolos distintos

  `t` `q` `-`
- Un número infinito de axiomas:

  x`t-q`*x* es un axioma siempre que *x* esté compuesto sólo de rayas (1 o más)
- Una regla única

  Suponga que *x*, *y* & *z* son cadenas particulares de rayas. Si se sabe que *x*`t`*y*`q`*z* es un teorema, entonces *x*`t`*y*`-q`*zx* también es un teorema

$$\left.\begin{array}{rcl} \texttt{t} & \Leftrightarrow & \times \\ \texttt{q} & \Leftrightarrow & = \\ \texttt{-} & \Leftrightarrow & 1 \\ \texttt{--} & \Leftrightarrow & 2 \\ \texttt{---} & \Leftrightarrow & 3 \\ \text{etc.} & & \end{array}\right\} \begin{array}{c} \texttt{---t--q------} \\ \Leftrightarrow \\ 3 \times 2 = 6 \end{array}$$

Si la interpretación está dentro de un meta-sistema formal que abarca al primero, tenemos un procedimiento **I**nteligente de decisión

Otra vez:

- Conjunto Recursivo de axiomas
  - Se puede verificar, en un tiempo finito, si cualquier enunciado es un axioma o no
- Sistema formal Completo
  - Todos los enunciados verdaderos se pueden obtener como teoremas a partir de los axiomas y las reglas de inferencia
  - Otra forma equivalente de completitud (Gödel, 1931)
    - Todo enunciado es demostrable o refutable a partir de los axiomas y las reglas de inferencia
- Sistema formal Consistente
  - Si puedo demostrar un enunciado, no puedo demostrar su negación
- $\Rightarrow$Incompleto
  - Existe al menos un enunciado indecidible (no se puede verificar en un tiempo finito si es un teorema o no)

Consistencia, completitud, interpretación

La consistencia con respecto a una interpretación particular dice si todos los teoremas son significativos dentro de nuestra interpretación.

La completitud con respecto a una interpretación particular dice si el sistema formal captura todos los enunciados posibles dentro de esa interpretación.

- **MIU**: Consistente pero incompleto (MU es indecidible), independiente de alguna interpretación
- ***pq***: Consistente y completo para la suma de naturales, pero no describe toda la aritmética
- ***pq* modificado**: Inconsistente para la suma de naturales, pero consistente para la relación ≥. Sin embargo, sólo puede representar "igual o mayor en 1": Aunque puede representar 3+5≥8 y 3+5≥7, no puede representar 3+5≥6. No es completo para la relación ≥, pero si es completo para la relación "igual o mayor en 1".
- ***tq***: Consistente y completo para la multiplicación de naturales, pero no describe toda la aritmética

Mediados del S. XIX

- Interés por encontrar nociones básicas y elementales que permitieran generar todas las otras nociones, y así reconstruir el edificio de las matemáticas desde bases sólidas e indiscutibles.
- Respuesta: ¡Teoría de conjuntos! Desde ella se puede definir el concepto de número, de relación, de función, de espacio vectorial, etc. con base en un concepto intuitivo: El conjunto.
- 1902: Frege estaba por terminar el trabajo (formidable) de refundar las matemáticas sobre la teoría de conjuntos, cuando Russell salió con un problemita: La paradoja de Russell.

Paradoja de Russell

- En general, los conjuntos no son elementos de si mismos, pero hay algunos que sí. Por ejemplo, el conjunto de los conceptos es un concepto y, por tanto, debe ser un elemento de sí mismo. El conjunto de todos los conjuntos también es un elemento de sí mismo. Entonces, se puede construir el conjunto de los conjuntos que no son elementos de sí mismos:

$$S = \left\{ X : X \notin X \right\}$$

- ¿$S \in S$? Si sí, no debería pertenecer. Si no, sí debería pertenecer.

- La definición intuitiva de conjunto era demasiado laxa y llevaba a contradicciones.

- **¡Debemos evitar la auto-referencia!**

Una página cualquiera (la 203, p.ej.) de *Principia Mathematica*

***20·23.** $\vdash : \hat{z}(\phi z) = \hat{z}(\psi z) \,.\, \hat{z}(\phi z) = \hat{z}(\chi z) \,.\supset.\, \hat{z}(\psi z) = \hat{z}(\chi z)$ [*20·21·22]

***20·24.** $\vdash : \hat{z}(\psi z) = \hat{z}(\phi z) \,.\, \hat{z}(\chi z) = \hat{z}(\phi z) \,.\supset.\, \hat{z}(\psi z) = \hat{z}(\chi z)$ [*20·21·22]

***20·25.** $\vdash :. \alpha = \hat{z}(\phi z) \,.\equiv_\alpha.\, \alpha = \hat{z}(\psi z) : \equiv .\, \hat{z}(\phi z) = \hat{z}(\psi z)$

Dem.

$\vdash .$ *10·1 $. \quad \supset \vdash :. \alpha = \hat{z}(\phi z) \,.\equiv_\alpha.\, \alpha = \hat{z}(\psi z) : \supset :$

$\hat{z}(\phi z) = \hat{z}(\phi z) \,.\equiv.\, \hat{z}(\phi z) = \hat{z}(\psi z) :$

[*20·2] $\supset : \hat{z}(\phi z) = \hat{z}(\psi z)$ (1)

$\vdash .$ *20·22 $. \quad \supset \vdash : \alpha = \hat{z}(\phi z) \,.\, \hat{z}(\phi z) = \hat{z}(\psi z) \,.\supset.\, \alpha = \hat{z}(\psi z) :$

[Exp.Comm] $\supset \vdash :. \hat{z}(\phi z) = \hat{z}(\psi z) \,.\supset : \alpha = \hat{z}(\phi z) \,.\supset.\, \alpha = \hat{z}(\psi z)$ (2)

$\vdash .$ *20·24 $. \quad \supset \vdash :. \hat{z}(\phi z) = \hat{z}(\psi z) \,.\, \alpha = \hat{z}(\psi z) \,.\supset.\, \alpha = \hat{z}(\phi z) :.$

[Exp] $\supset \vdash :. \hat{z}(\phi z) = \hat{z}(\psi z) \,.\supset : \alpha = \hat{z}(\psi z) \,.\supset.\, \alpha = \hat{z}(\phi z)$ (3)

$\vdash . (2) . (3) . \quad \supset \vdash :. \hat{z}(\phi z) = \hat{z}(\psi z) \,.\supset : \alpha = \hat{z}(\phi z) \,.\equiv.\, \alpha = \hat{z}(\psi z) :.$

[*10·11·21] $\supset \vdash :. \hat{z}(\phi z) = \hat{z}(\psi z) \,.\supset : \alpha = \hat{z}(\phi z) \,.\equiv_\alpha.\, \alpha = \hat{z}(\psi z)$ (4)

$\vdash . (1) . (4) . \quad \supset \vdash .$ Prop

***20·3.** $\vdash : x \,\epsilon\, \hat{z}(\psi z) \,.\equiv.\, \psi x$

Dem.

$\vdash .$ *20·1 $. \supset$

$\vdash :: x \,\epsilon\, \hat{z}(\psi z) \,.\equiv :. (\exists \phi) :. \psi y \,.\equiv_y.\, \phi ! y : x \,\epsilon\, (\phi ! \hat{z}) :.$

[(*20·02)] $\equiv :. (\exists \phi) :. \psi y \,.\equiv_y.\, \phi ! y : \phi ! x :.$

[*10·43] $\equiv :. (\exists \phi) :. \psi y \,.\equiv_y.\, \phi ! y : \psi x :.$

[*10·35] $\equiv :. (\exists \phi) : \psi y \,.\equiv_y.\, \phi ! y :. \psi x :.$

[*12·1] $\equiv :. \psi x :: \supset \vdash .$ Prop

***20·31.** $\vdash :. \hat{z}(\psi z) = \hat{z}(\chi z) \,.\equiv : x \,\epsilon\, \hat{z}(\psi z) \,.\equiv_x.\, x \,\epsilon\, \hat{z}(\chi z)$ [*20·15·3]

***20·32.** $\vdash .\, \hat{x}\{x \,\epsilon\, \hat{z}(\phi z)\} = \hat{z}(\phi z)$ [*20·3·15]

***20·33.** $\vdash :. \alpha = \hat{z}(\phi z) \,.\equiv : x \,\epsilon\, \alpha \,.\equiv_x.\, \phi x$

Dem.

$\vdash .$ *20·31 $. \quad \supset \vdash :. \alpha = \hat{z}(\phi z) \,.\equiv : x \,\epsilon\, \alpha \,.\equiv_x.\, x \,\epsilon\, \hat{z}(\phi z)$ (1)

$\vdash . (1) .$ *20·3 $. \supset \vdash .$ Prop

***20·34.** $\vdash :. x = y \,.\equiv : x \,\epsilon\, \alpha \,.\supset_\alpha.\, y \,\epsilon\, \alpha$

Dem.

$\vdash .$ *4·2 . (*20·07) $. \supset \vdash :. x \,\epsilon\, \alpha \,.\supset_\alpha.\, y \,\epsilon\, \alpha : \equiv : x \,\epsilon\, \hat{z}(\phi ! z) \,.\supset_\phi.\, y \,\epsilon\, \hat{z}(\phi ! z) :$

[*20·3] $\equiv : \phi ! x \,.\supset_\phi.\, \phi ! y :$

[*13·1] $\equiv : x = y :. \supset \vdash .$ Prop

Los números naturales (Peano)

$$\mathbb{N}(0)$$

$$\forall x(\mathbb{N}(x) \Rightarrow \mathbb{N}(S(x)))$$

$$\neg\exists x(\mathbb{N}(x) \wedge S(x) = 0)$$

$$\forall x \forall y\big((\mathbb{N}(x) \wedge \mathbb{N}(y) \wedge S(x) = S(y)) \Rightarrow x = y\big)$$

$$\forall \varphi\Big(\varphi(0) \wedge \forall x\big((\varphi(x) \Rightarrow \varphi(S(x))\big) \Rightarrow \forall x \varphi(x)\Big)$$

La suma de números naturales

$\forall x\ (x + 0 = x)$

$\forall x \forall y\ (\ x + \mathrm{S}(y) = \mathrm{S}(x + y)\ )$

La multiplicación de números naturales

$\forall x\ (x \cdot 0 = 0)$

$\forall x \forall y\ (\ x \cdot \mathrm{S}(y) = x + (x \cdot y))$

En segundo de primaria no nos deberían torturar con las tablas de multiplicar: Si sólo nos enseñaran esto, las podríamos deducir ☺

Consistencia y completitud de la teoría de los números naturales

1. **Podemos codificar cada enunciado y cada secuencia de enunciados como un número natural, de una manera mecánica y sistemática (Numeración de Gödel).**

Existe una fórmula con dos variables libres, $\omega(x,y)$, tal que el enunciado $\omega(m,n)$ es un teorema si y sólo si m es el número de Gödel de la fórmula con una variable libre, $\varphi(x)$, y n es el número de Gödel de la demostración de $\varphi(m)$. P. ej.,

La fórmula $\varphi(x)=(\exists y)y^2=x$ dice que x es un cuadrado perfecto

Esta fórmula tiene un número Gödel, m

n es el número Gödel de la demostración de $\varphi(m)$ [el número Gödel de $\varphi(x)$ es un cuadrado perfecto]

Ahora construyamos $\psi(x)=(\forall y)(\sim\omega(x,y))$. $\Rightarrow$ $\psi(m)$ dice que no hay una demostración de $\varphi(m)$

(En realidad dice que no existe un y que sea el número Gödel de la demostración de $\varphi(m)$)

Consistencia y completitud de la teoría de los números naturales

2. **Autoreferencia: Podemos Construir un enunciado que niegue su demostrabilidad**

Sea p el número Gödel de ψ ,y sea β el enunciado $\psi(p)$. Entonces β es un enunciado que niega su propia demostrabilidad

Como $\psi(p)$ dice que no hay una demostración de $\varphi(p)$, donde φ es el enunciado con número Gödel p, β quiere decir que no hay una demostración de $\psi(p)$. ¡Pero $\psi(p)$ es β!

β afirma que β no es demostrable, lo cual TIENE que ser cierto porque una demostración de β sería una demostración de que no se puede demostrar.

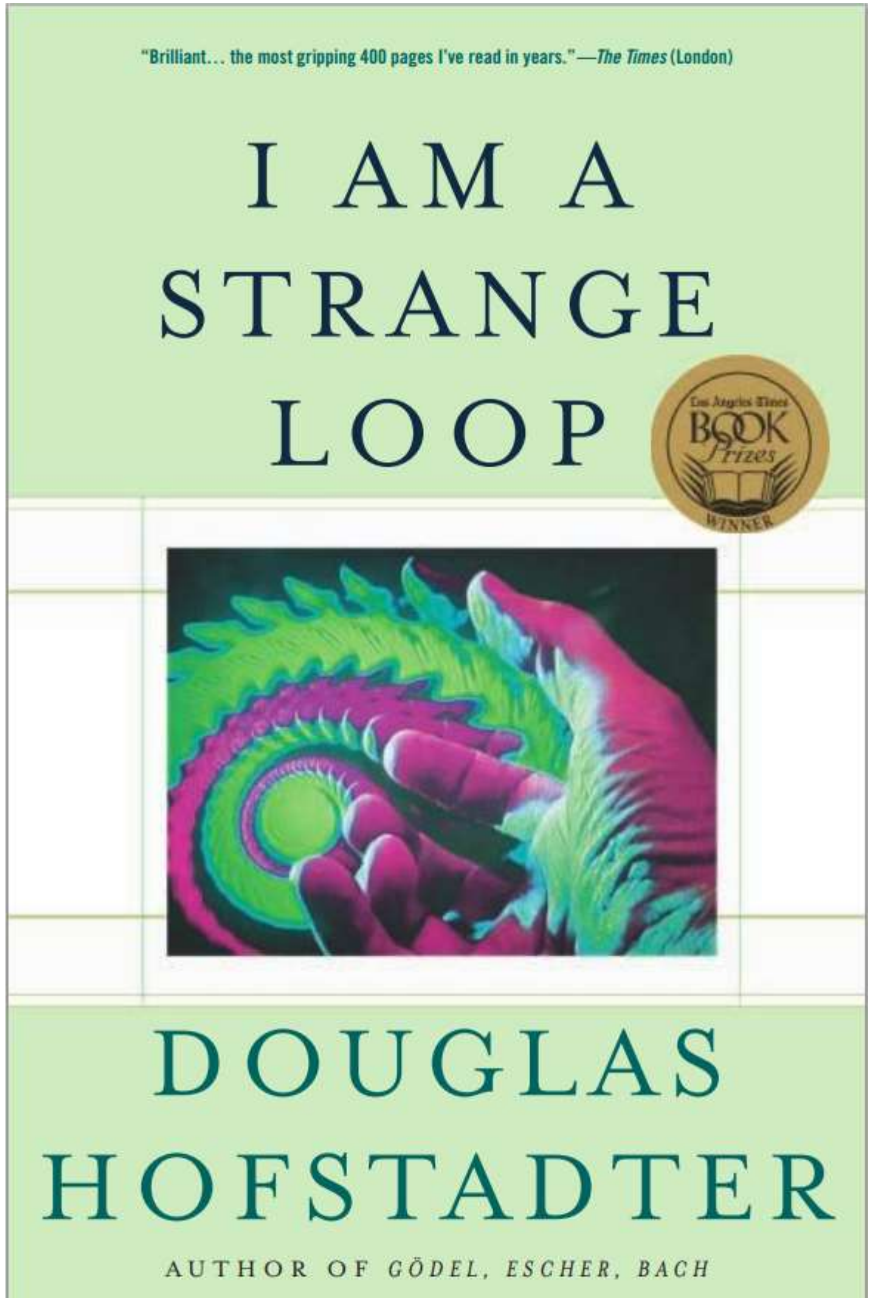
Esta inferencia lógica (demostración) es un criterio Inteligente de decisión: Esta fuera del sistema formal de Peano.

Teorema de incompletitud de Gödel

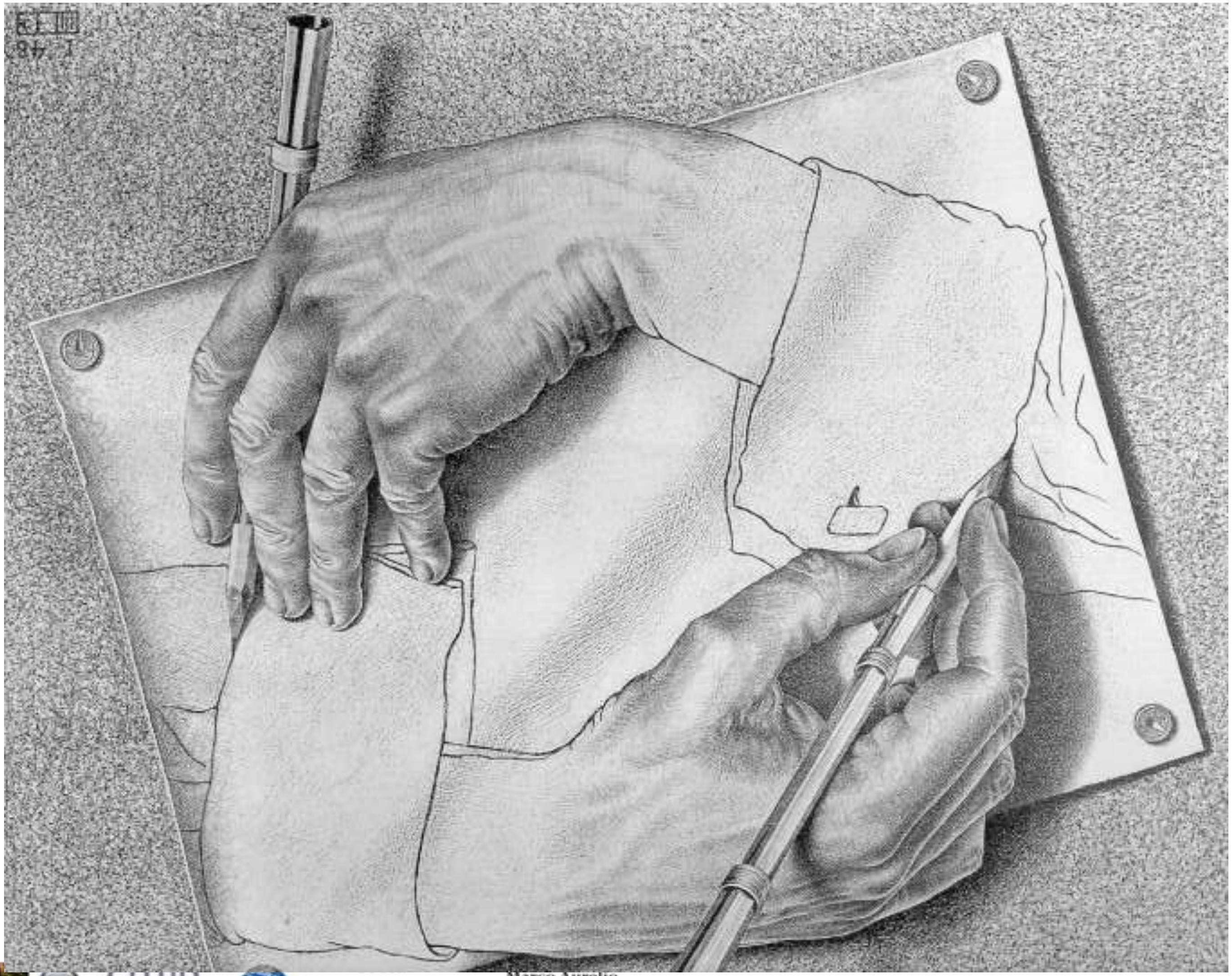
- Todo sistema axiomático consistente y recursivo **para la aritmética** tiene enunciados indecidibles. En particular, puede exhibirse un enunciado verdadero pero no demostrable dentro del sistema.

- No podemos evitar la auto-referencia

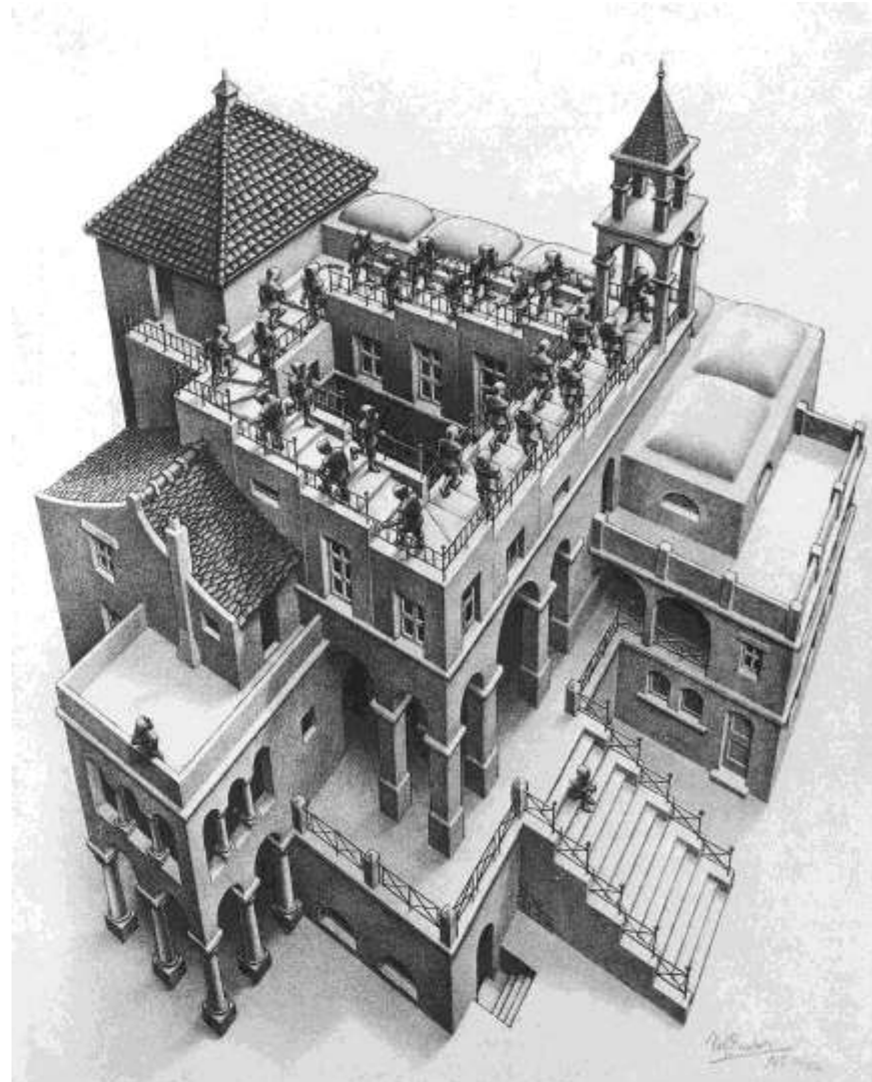
“Yo no soy demostrable”


"Brilliant… the most gripping 400 pages I've read in years."—The Times (London)
I AM A STRANGE LOOP
Los Angeles Times BOOK Prizes WINNER
DOUGLAS HOFSTADTER
AUTHOR OF GÖDEL, ESCHER, BACH

Douglas R. Hofstadter,
"Gödel, Escher, Bach: an Eternel Golden Braid",
Basic Books, 1979



Douglas R. Hofstadter,
“Gödel, Escher, Bach: an Eternel Golden Braid”,
Basic Books, 1979



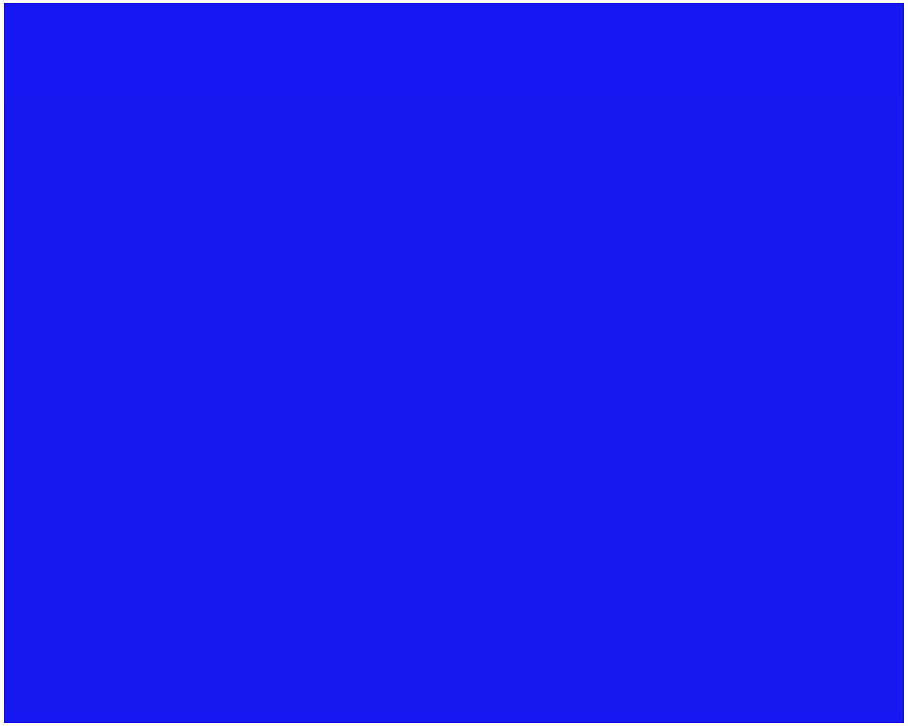
Douglas R. Hofstadter,
"Gödel, Escher, Bach: an Eternel Golden Braid",
Basic Books, 1979

Douglas R. Hofstadter,
“Gödel, Escher, Bach: an Eternel Golden Braid”,
Basic Books, 1979

http://www.youtube.com/watch?v=9WHdyG9mJaI

Douglas R. Hofstadter,
"Gödel, Escher, Bach: an Eternel Golden Braid",
Basic Books, 1979

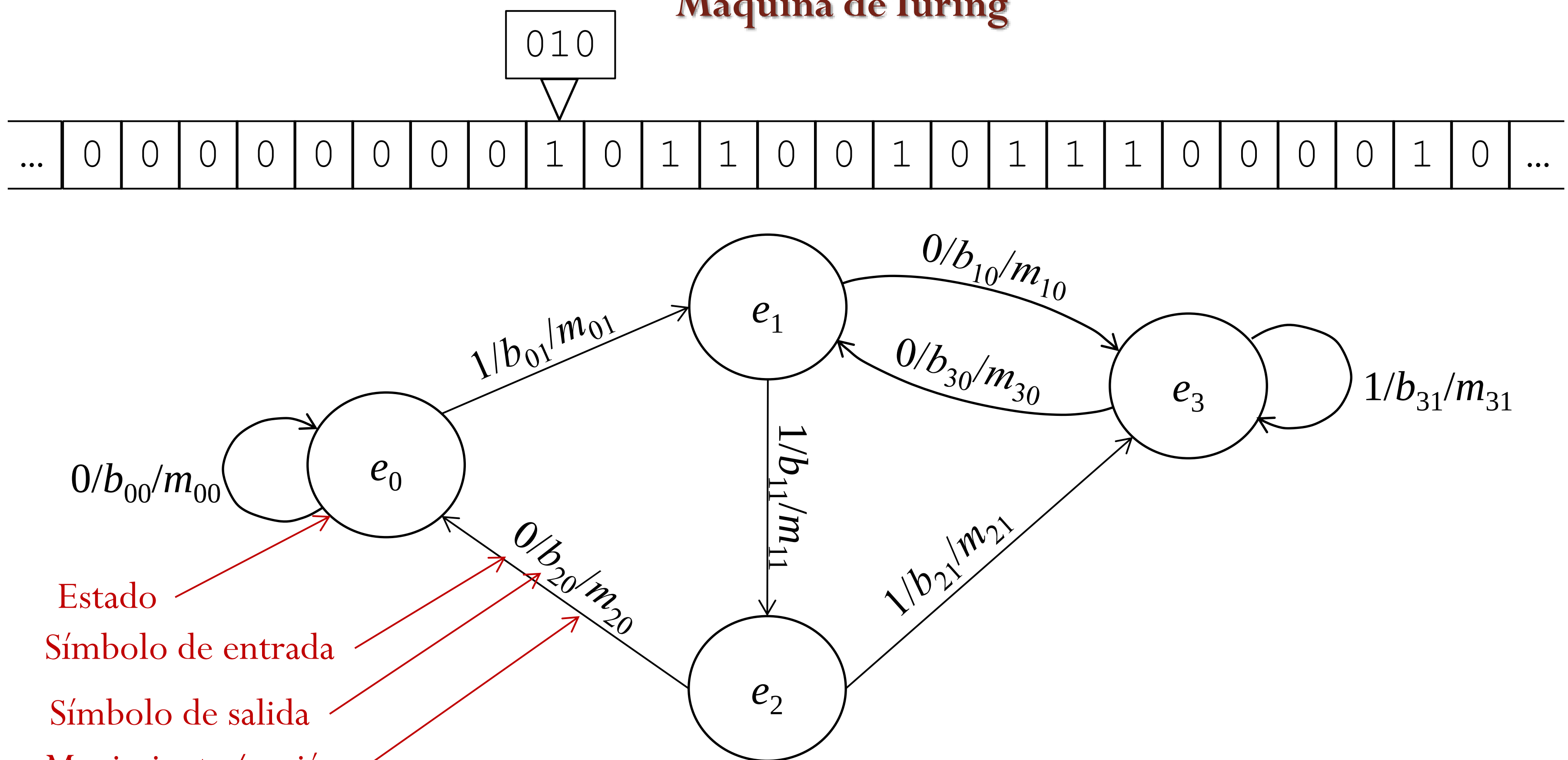
http://www.youtube.com/watch?v=cwhLDLQLI44



MIU: del modo Mecánico al modo Inteligente hasta el modo Un-mode

El modo **M**ecánico consiste en manipular símbolos carentes de significado de acuerdo con reglas formales

Máquina de Turing

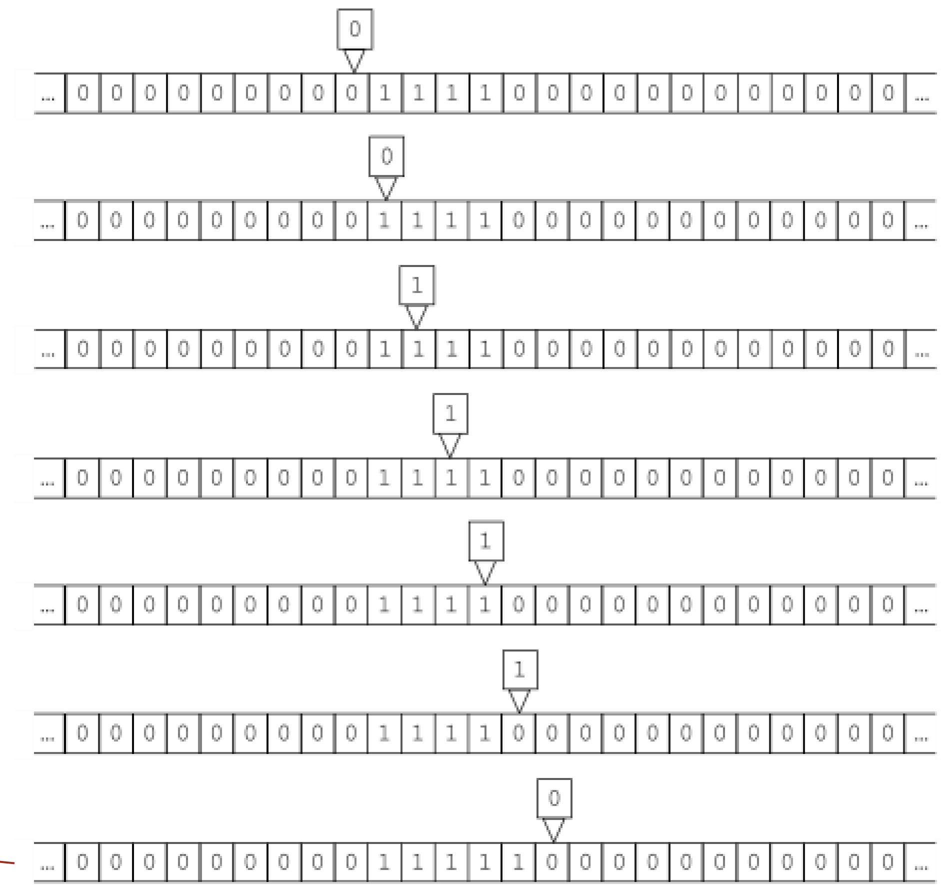


El sistema *pq*:

e[*n*]	*b*[*n*]	*e*[*n*+1]	*b*[*n*+1]	*m*[*n*]
0	0	0	0	R
0	1	1	1	R
1	0	0	1	D-H
1	1	1	1	D

Toma un número en "código unario" y le suma 1

¿Cuánto es 4+1?



----p-q-----

```
function salida = turing(tabla, entrada)
    % Salida = Turing(tabla, entrada)
    % tabla(:,3) describe "el programa" de la máquina de turing:
    %     Si en el estado e lee el bit b, busca qué hacer asi:
    %         tabla(2*e+b+1,1) = siguiente estado
    %         tabla(2*e+b+1,2) = bit que escribe en la cinta
    %         tabla(2*e+b+1,3) = acción: 0 = se mueve a la derecha
    %                                    1 = se mueve a la izquierda
    %                                    2 = se detiene
    % entrada[] es la lista de bits que entran a la máquina
    entrada = [zeros(100,1); entrada(:); zeros(100,1)]; % Supone una cinta infinita (100 ceros a cada lado)
    estado = 0; % la máquina empieza en el estado 0.
    n = 101;    % la máquina empieza en el primer bit de entrada[],
    minimo = n; % Esta es la posición más a la izquierda que se ha visitado
    while(1)    % Sale cuando llegue a la acción ALTO
        b = entrada(n);                 % bit leído
        indice = 2*estado+b+1;          % indice en la tabla
        estado = tabla(indice,1);       % siguiente estado
        entrada(n) = tabla(indice,2);   % bit que se escribe en la cinta
        accion = tabla(indice,3);       % acción a seguir
        switch accion
            case 0                      % moverse a la derecha
                n = n + 1;
                if n>length(entrada)
                    entrada = [entrada; 0];
                end
            case 1                      % moverse a la izquierda
                n = n - 1;
                if n<1
                    n = 1;
                    entrada = [0; entrada];
                end
                if n<minimo, minimo=n; end  % recuerda la posición más a la izquierda que se ha visitado
            case 2                     % ALTO : Terminó
                 break
        end
    end
    salida = entrada(minimo:n); % Retorna la cinta a la izquierda de la cabeza lectora
```

4+1=?

```
>> tabla = [0 0 0;
            1 1 0;
            0 1 2;
            1 1 0];
 entrada = [1 1 1 1]';
 salida = turing(tabla, entrada)'

0  0  0  0  0  0  0  0  0  0  0  0 - 1 - 1  1  1  0  0  0  0  0  0  0  0  0   :  0
0  0  0  0  0  0  0  0  0  0  0  1 - 1 - 1  1  0  0  0  0  0  0  0  0  0  0   :  1
0  0  0  0  0  0  0  0  0  0  1  1 - 1 - 1  0  0  0  0  0  0  0  0  0  0  0   :  1
0  0  0  0  0  0  0  0  0  1  1  1 - 1 - 0  0  0  0  0  0  0  0  0  0  0  0   :  1
0  0  0  0  0  0  0  0  1  1  1  1 - 0 - 0  0  0  0  0  0  0  0  0  0  0  0   :  1
0  0  0  0  0  0  0  0  1  1  1  1 - 1 - 0  0  0  0  0  0  0  0  0  0  0  0   :  0

salida =

     1     1     1     1     1
```

El sistema *tq*:

$e[n]$	$b[n]$	$e[n+1]$	$b[n+1]$	$m[n]$
000	0	000	0	R
000	1	001	0	R
001	0	010	1	L
001	1	001	1	R
010	0	011	0	R
010	1	100	0	R
011	0	000	1	R-H
011	1	011	1	R
100	0	101	1	L
100	1	1100	1	R
101	0	010	1	L
101	1	101	1	l

Toma un número en "código unario" y lo multiplica por 2

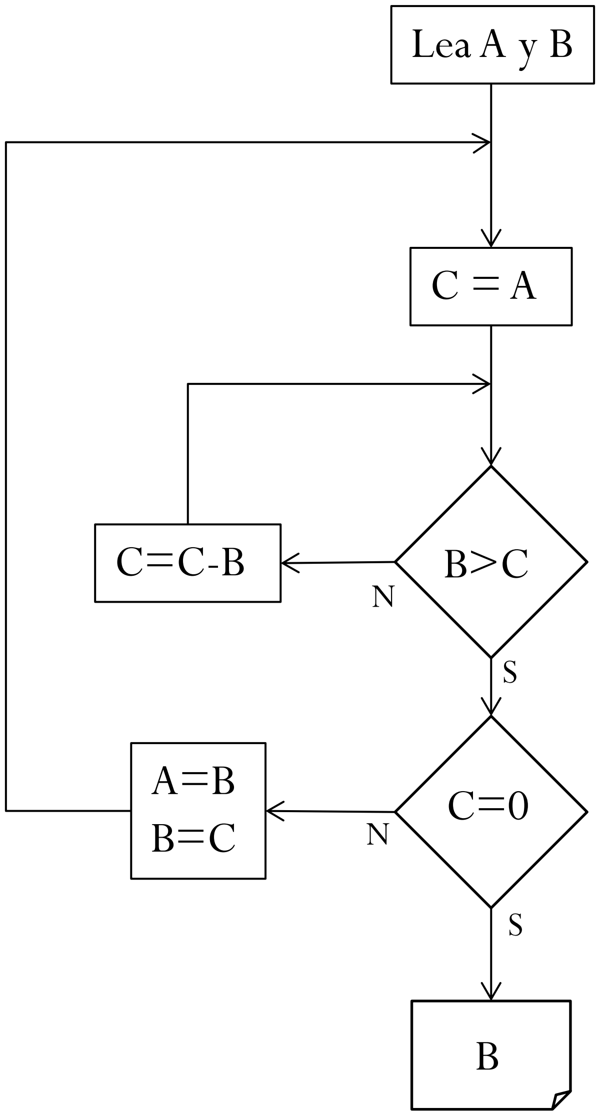
3×2=?

```
 tabla = [0 0 0;    1 0 0;    2 1 1;    1 1 0;
          3 0 0;    4 0 0;    0 1 2;    3 1 0;
          5 1 1;    4 1 0;    2 1 1;    5 1 1];
entrada = [1 1 1]';
salida = turing(tabla, entrada)'
```

```
0 0 0 0 0 0 0 0 0 0 0 0 - 1 - 1 1 0 0 0 0 0 0 0 0 0 0 : 0
0 0 0 0 0 0 0 0 0 0 0 0 - 1 - 1 0 0 0 0 0 0 0 0 0 0 0 : 1
0 0 0 0 0 0 0 0 0 0 0 1 - 1 - 0 0 0 0 0 0 0 0 0 0 0 0 : 1
0 0 0 0 0 0 0 0 0 0 1 1 - 0 - 0 0 0 0 0 0 0 0 0 0 0 0 : 1
0 0 0 0 0 0 0 0 0 0 0 1 - 1 - 1 0 0 0 0 0 0 0 0 0 0 0 : 2
0 0 0 0 0 0 0 0 0 0 1 0 - 1 - 0 0 0 0 0 0 0 0 0 0 0 0 : 4
0 0 0 0 0 0 0 0 0 1 0 1 - 0 - 0 0 0 0 0 0 0 0 0 0 0 0 : 4
0 0 0 0 0 0 0 0 0 0 1 0 - 1 - 1 0 0 0 0 0 0 0 0 0 0 0 : 5
0 0 0 0 0 0 0 0 0 0 0 1 - 0 - 1 1 0 0 0 0 0 0 0 0 0 0 : 5
0 0 0 0 0 0 0 0 0 0 0 0 - 1 - 1 1 1 0 0 0 0 0 0 0 0 0 : 2
0 0 0 0 0 0 0 0 0 0 0 0 - 1 - 1 1 0 0 0 0 0 0 0 0 0 0 : 4
0 0 0 0 0 0 0 0 0 0 0 1 - 1 - 1 0 0 0 0 0 0 0 0 0 0 0 : 4
0 0 0 0 0 0 0 0 0 0 1 1 - 1 - 0 0 0 0 0 0 0 0 0 0 0 0 : 4
0 0 0 0 0 0 0 0 0 1 1 1 - 0 - 0 0 0 0 0 0 0 0 0 0 0 0 : 4
0 0 0 0 0 0 0 0 0 0 1 1 - 1 - 1 0 0 0 0 0 0 0 0 0 0 0 : 5
0 0 0 0 0 0 0 0 0 0 0 1 - 1 - 1 1 0 0 0 0 0 0 0 0 0 0 : 5
0 0 0 0 0 0 0 0 0 0 0 0 - 1 - 1 1 1 0 0 0 0 0 0 0 0 0 : 5
0 0 0 0 0 0 0 0 0 0 0 0 - 0 - 1 1 1 1 0 0 0 0 0 0 0 0 : 5
0 0 0 0 0 0 0 0 0 0 0 0 - 0 - 1 1 1 1 1 0 0 0 0 0 0 0 : 2
0 0 0 0 0 0 0 0 0 0 0 0 - 1 - 1 1 1 1 0 0 0 0 0 0 0 0 : 3
0 0 0 0 0 0 0 0 0 0 0 1 - 1 - 1 1 1 0 0 0 0 0 0 0 0 0 : 3
0 0 0 0 0 0 0 0 0 0 1 1 - 1 - 1 1 0 0 0 0 0 0 0 0 0 0 : 3
0 0 0 0 0 0 0 0 0 1 1 1 - 1 - 1 0 0 0 0 0 0 0 0 0 0 0 : 3
0 0 0 0 0 0 0 1 1 1 1 1 - 1 - 0 0 0 0 0 0 0 0 0 0 0 0 : 3
0 0 0 0 0 0 0 1 1 1 1 1 - 0 - 0 0 0 0 0 0 0 0 0 0 0 0 : 3
0 0 0 0 0 0 0 1 1 1 1 1 - 1 - 0 0 0 0 0 0 0 0 0 0 0 0 : 0

salida =   1    1    1    1    1    1
```

Inclusive el algoritmo de Euclides para encontrar el máximo común divisor entre dos números:



e[*n*]	*b*[*n*]	*e*[*n*+1]	*b*[*n*+1]	*m*[*n*]
0000	0	0000	0	R
0000	1	0001	1	L
0001	0	0010	1	R
0001	1	0001	1	L
0010	0	1010	0	R
0010	1	0011	0	R
0011	0	0100	0	R
0011	1	0011	1	R
0100	0	0100	0	R
0100	1	0101	0	R
0101	0	0111	0	L
0101	1	0110	1	L
0110	0	0110	0	L
0110	1	0001	1	L
0111	0	0111	0	L
0111	1	1000	1	L
1000	0	1001	0	L
1000	1	1000	1	L
1001	0	0010	0	R
1001	1	0001	1	L
1010	0	0000	0	R-H
1010	1	1010	1	R

MCD(8,4)=?

```
tabla = [ 0 0 0;     1 1 1;     2 1 0;     1 1 1;    10 0 0;     3 0 0;
          4 0 0;     3 1 0;     4 0 0;     5 0 0;     7 0 1;     6 1 1;
          6 0 1;     1 1 1;     7 0 1;     8 1 1;     9 0 1;     8 1 1;
          2 0 0;     1 1 1;     0 0 2;    10 1 0];
entrada = [1 1 1 1 1 1 1 1 0 1 1 1 1]';
salida = turing(tabla, entrada, 1)'
```

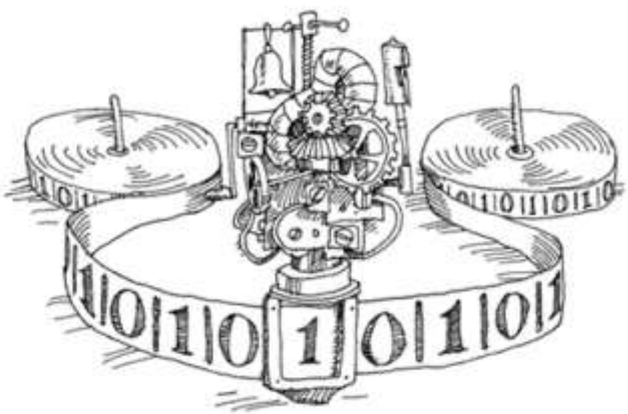
```
0 0 0 0 0 0 0 0 0 0 0 0 - 1 - 1 1 1 1 1 1 1 0 1 1 1 1 : 0
0 0 0 0 0 0 0 0 0 0 0 0 - 0 - 1 1 1 1 1 1 1 1 0 1 1 1 : 1
0 0 0 0 0 0 0 0 0 0 0 1 - 1 - 1 1 1 1 1 1 1 0 1 1 1 1 : 2
0 0 0 0 0 0 0 0 0 0 1 0 - 1 - 1 1 1 1 1 1 0 1 1 1 1 0 : 3
0 0 0 0 0 0 0 0 0 1 0 1 - 1 - 1 1 1 1 1 0 1 1 1 1 0 0 : 3
0 0 0 0 0 0 0 0 1 0 1 1 - 1 - 1 1 1 1 0 1 1 1 1 0 0 0 : 3
0 0 0 0 0 0 0 1 0 1 1 1 - 1 - 1 1 1 0 1 1 1 1 0 0 0 0 : 3
0 0 0 0 0 0 1 0 1 1 1 1 - 1 - 1 1 0 1 1 1 1 0 0 0 0 0 : 3
0 0 0 0 0 1 0 1 1 1 1 1 - 1 - 1 0 1 1 1 1 0 0 0 0 0 0 : 3
0 0 0 0 1 0 1 1 1 1 1 1 - 1 - 0 1 1 1 1 0 0 0 0 0 0 0 : 3
0 0 0 1 0 1 1 1 1 1 1 1 - 0 - 1 1 1 1 0 0 0 0 0 0 0 0 : 3
0 0 1 0 1 1 1 1 1 1 1 0 - 1 - 1 1 1 0 0 0 0 0 0 0 0 0 : 4
0 1 0 1 1 1 1 1 1 1 0 0 - 1 - 1 1 0 0 0 0 0 0 0 0 0 0 : 5
0 0 1 0 1 1 1 1 1 1 1 0 - 0 - 1 1 1 0 0 0 0 0 0 0 0 0 : 6
0 0 0 1 0 1 1 1 1 1 1 1 - 0 - 0 1 1 1 0 0 0 0 0 0 0 0 : 6
0 0 0 0 1 0 1 1 1 1 1 1 - 1 - 0 0 1 1 1 0 0 0 0 0 0 0 : 6
0 0 0 0 0 1 0 1 1 1 1 1 - 1 - 1 0 0 1 1 1 0 0 0 0 0 0 : 1
0 0 0 0 0 0 1 0 1 1 1 1 - 1 - 1 1 0 0 1 1 1 0 0 0 0 0 : 1
0 0 0 0 0 0 0 1 0 1 1 1 - 1 - 1 1 1 0 0 1 1 1 0 0 0 0 : 1
0 0 0 0 0 0 0 0 1 0 1 1 - 1 - 1 1 1 1 0 0 1 1 1 0 0 0 : 1
0 0 0 0 0 0 0 0 0 1 0 1 - 1 - 1 1 1 1 1 0 0 1 1 1 0 0 : 1
0 0 0 0 0 0 0 0 0 0 1 0 - 1 - 1 1 1 1 1 1 0 0 1 1 1 0 : 1
 ...           ...         ...        ...       ...      ...
0 0 0 0 0 0 0 0 0 0 0 1 - 1 - 1 1 0 0 0 0 0 0 0 0 0 0 : 10
0 0 0 0 0 0 0 0 0 0 1 1 - 1 - 1 0 0 0 0 0 0 0 0 0 0 0 : 10
0 0 0 0 0 0 0 0 0 1 1 1 - 1 - 0 0 0 0 0 0 0 0 0 0 0 0 : 10
0 0 0 0 0 0 0 0 1 1 1 1 - 0 - 0 0 0 0 0 0 0 0 0 0 0 0 : 10
0 0 0 0 0 0 0 0 1 1 1 1 - 0 - 0 0 0 0 0 0 0 0 0 0 0 0 : 0

salida =    0    1    1    1    1    0
```

MCD(4,8) = 4

Debe quedar claro hasta aquí que

- Cualquier operación aritmética puede ser calculada efectivamente por una máquina de Turing particular
- Podemos construir máquinas de Turing sin un cálculo pre-especificado pero con instrucciones incluidas en la cinta para simular una máquina de Turing particular (máquina universal de Turing).
- Añadamos algunas interfaces de entrada/salida y un sistema operativo que vaya seleccionando las máquinas de Turing a simular y… ¡listo!: Nuestros computadores modernos son sofisticadas máquinas universales de Turing



≡

Tesis de Church–Turing

Una función es algorítmicamente computable si y sólo si es computable por una máquina de Turing (*)

(*) A mi me parece más una definición que una conjetura…

Otros modelos de computación

- Cálculo Lambda (Church)
- Lógica Combinatoria (Schönfinkel and Curry)
- Funciones recursivas Mu (Kleene)
- Máquina registradora (Wang)
- P" (Böhm)
- Etc.

En favor de la tesis de Church-Turing:

Se ha demostrado que todos ellos son esencialmente equivalentes a la máquina de Turing

"computador universal"

¿ "As far as we know, no device built in the physical universe can have any more computational power than a Turing machine. To put it more precisely, any computation that can be performed by any physical computing device can be performed by any universal computer, as long as the latter has sufficient time and memory." ?

D. Hillis, The Pattern on the Stone, Basic Books, New York, 1998

MIU: del modo Mecánico al modo Inteligente hasta el modo Un-mode

Modo Mecánico

Una máquina de Turing puede demostrar que MUI es un teorema en el sistema MIU:
MI → MII → MIIII → MUI → Para

Una máquina de Turing no puede demostrar que MU es un teorema en el sistema MIU:
MI → MII → MIIII → MUI → … (No para)

Modo Inteligente

Una máquina de Turing puede preguntarse si la anterior máquina de Turing parará ante el problema MU del sistema MIU (¿el número de Ies es divisible por 3?)

Modo Un-mode

¿Una máquina de Turing puede preguntarse si ella misma parará ante un problema computacional particular?

Modo Mecánico

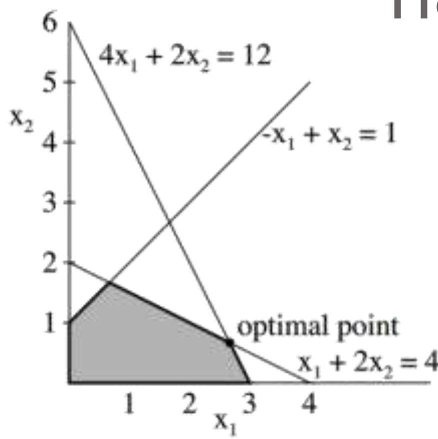
Modo Inteligente

Modo Un-mode

MIU: del modo **M**ecánico al modo **I**nteligente hasta el modo **U**n-mode

$$\underset{x_1, x_2}{\text{maximizar}} \quad (x_1 + x_2)$$

$$\text{sujeto a} \quad x_1 \geq 0$$
$$x_2 \geq 0$$
$$x_1 + 2x_2 \leq 4$$
$$2x_1 + x_2 \leq 6$$
$$-x_1 + x_2 \leq 1$$



Ingeniería Clásica: Modo **M**ecánico

Control cooperativo
Sistemas cognitivos
Auto-organización guíada
Ingeniería morfogenética

Ingeniería de sistemas complejos: Modo **U**n-mode

$$\min \sum_{i \in V} \sum_{j \in V} c_{ij} \sum_{k=1}^{K} x_{ijk}$$

$$\sum_{k=1}^{K} y_{0k} \leq K, \quad k = 1, \ldots, K$$

$$\sum_{j \in V} x_{ijk} = \sum_{j \in V} x_{jik} = y_{ik}, \quad \forall i \in V, k = 1, \ldots, K$$

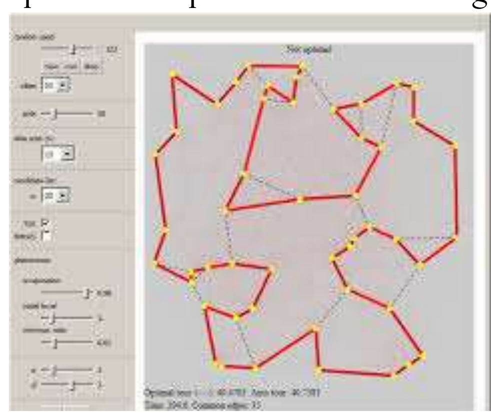
$$\sum_{i \in S} \sum_{j \notin S} x_{ijk} \geq y_{hk},$$

$$\forall S \subseteq V \setminus \{0\}, h \in S, k = 1, \ldots, K$$

$$\sum_{k=1}^{K} y_{ik} \leq 1, \quad \forall i \in V \setminus \{0\}$$

$$\sum_{k=1}^{K} z_{ilk} \leq s_{li}, \quad \forall l \in S, \forall i \in V$$

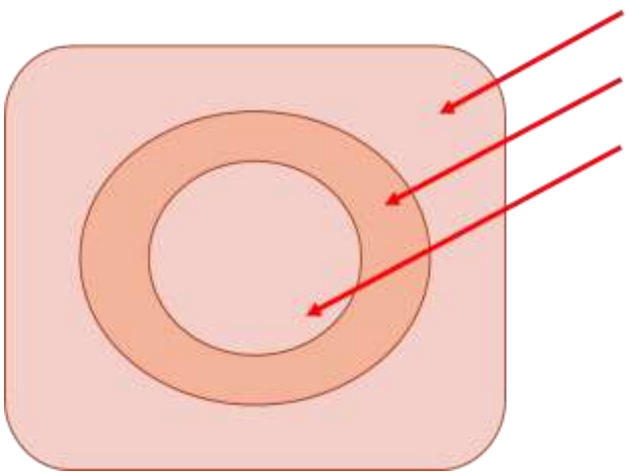
Optimización por colonia de hormigas



Ingeniería Clásica BioInspirada: Modo **I**nteligente

CONCLUSIONES

Conclusiones: Lo verdadero y lo demostrable



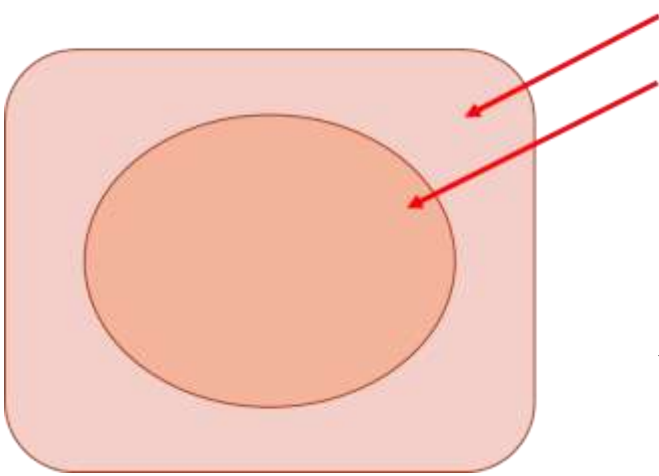
Enunciados

Enunciados verdaderos

Enunciados demostrables

En algún sistema formal particular,
bajo alguna interpretación particular

Este es el caso de cualquier sistema axiomático recursivo y consistente que tenga suficiente aritmética



Enunciados

Enunciados verdaderos =

Enunciados demostrables

En algún sistema formal particular,
bajo alguna interpretación particular

Conclusiones: Algunos Mitos y Realidades con respecto a los teoremas de Gödel

- Mito: Los teoremas de Gödel establecen un límite a las pretensiones de la razón humana.
- Realidad: Los teoremas de Gödel establecen un límite para la potencialidad del formalismo puro en matemáticas.

Tal vez los poderes del razonamiento humano sean limitados, pero los teoremas de Gödel no son ninguna evidencia de ese hecho porque los sistemas formales no modelan el razonamiento humano.

Los teoremas de Gödel ni siquiera establecen un límite para las capacidades de modelamiento de la matemática, porque los sistemas formales no son un modelo exhaustivo del razonamiento matemático

Conclusiones: Algunos Mitos y Realidades con respecto a los teoremas de Gödel

- Mito: Los teoremas de Gödel dicen que ninguna verdad puede establecerse de manera definitiva.
- Realidad: Los teoremas de Gödel no se preocupan por la verdad, sino por la insuficiencia de los métodos axiomáticos para demostrar la totalidad de los enunciados verdaderos.

Los teoremas de Gödel son verdades establecidas de manera definitiva

Conclusiones: Algunos Mitos y Realidades con respecto a los teoremas de Gödel

- Mito: Los teoremas de Gödel dicen que, ni siquiera en las matemáticas, puede haber total certidumbre.
- Realidad: Los teoremas de Gödel no ponen en duda ninguno de los resultados matemáticos ya establecidos (ni los que se establezcan en adelante). Sólo revelan una limitación de los métodos axiomáticos para comprobar algunos resultados verdaderos.

En las matemáticas se tiene tanta certeza de los teoremas de Gödel como de que 2+2=4

Conclusiones: Algunos Mitos y Realidades con respecto a los teoremas de Gödel

- Mito: El teorema de inconsistencia de Gödel dice que ninguna teoría puede ser consistente.
- Realidad: El teorema de inconsistencia de Gödel dice que la consistencia de algunas teorías recursivas no puede ser demostrada dentro de esas mismas teorías.

En su tesis doctoral, Gödel demostró que la lógica de predicados es una teoría consistente

Conclusiones: Algunos Mitos y Realidades con respecto a los teoremas de Gödel

- Mito: Los teoremas de Gödel dicen que ninguna teoría puede ser a la vez consistente y completa.
- Realidad: Los teoremas de Gödel dicen que una teoría "*que contenga suficiente aritmética*" no puede ser a la vez consistente y completa.

El cálculo de predicados es una teoría consistente y completa

"*que contenga suficiente aritmética*" : que permita demostrar dentro del sistema todas las verdades finitistas de la aritmética.

Conclusiones: Algunos Mitos y Realidades con respecto a los teoremas de Gödel

- Mito: Los teoremas de Gödel dicen que toda teoría de la aritmética es incompleta.
- Realidad: Los teoremas de Gödel dicen que toda teoría **recursiva** de la aritmética es incompleta.

El conjunto de todos los enunciados verdaderos de $\mathbb{N}$, $T(\mathbb{N})$ es una axiomatización completa (y trivial) de la aritmética, pero no es recursiva

Conclusiones: Algunos Mitos y Realidades con respecto a los teoremas de Gödel

- Mito: Los teoremas de Gödel dicen que toda teoría recursiva es incompleta.
- Realidad: Los teoremas de Gödel dicen que toda teoría recursiva, en la que pueda definirse la aritmética, es incompleta.

Sólo en la aritmética puede reproducirse el argumento de la demostración de incompletitud

($\mathbb{C}$,+,*) es una teoría matemática recursiva y completa

Conclusiones: Algunos Mitos y Realidades con respecto a los teoremas de Gödel

- Mito: Los teoremas de Gödel no tienen ninguna incidencia en las matemáticas.
- Realidad: ¡Tampoco! Los teoremas de Gödel no han tenido efectos pragmáticos en el razonamiento matemático, ni ha puesto en duda ninguno de esos métodos de razonamiento. Pero los teoremas de Gödel sí inauguraron la rama de la decidibilidad matemática y su demostración introdujo una idea fundamental: la autoreferencia.

“YO NO SOY DEMOSTRABLE”

¡GRACIAS!