Informática y Matemática

Juan Grompone

2006

grompone@interfase.com.uy

Temario

- Algunas nociones históricas.
- La numerabilidad de los conjuntos infinitos.
- Turing y sus resultados.
- La informática ayuda a la matemática
- Las limitaciones de la informática.
- ¿Una nueva matemática?

Algunas nociones históricas

Los problemas de Hilbert

- David Hilbert (1862, 1943) propuso, en el congreso de París de 1900, 23 problemas que suponía que contenían el futuro de la matemática en el siglo XX.
- Entre ellos estaba el problema de la consistencia lógica de la matemática y el problema de la prueba automática de los teoremas.

Hilbert estaba en buen camino

- Con los “Principia Mathematica” (1910-1913), Russell y Whitehead intentaron la formalización de la matemática.
- En 1931 Kurt Gödel demostró su célebre teorema de “incompletitud”.
- En 1936 Alan Turing demostró su resultado sobre los autómatas universales.
- En 1946 entra en operación ENIAC, la primera computadora electrónica. Comienza una revolución.

Parámetros de crecimiento

- Las computadoras crecieron a una velocidad sorprendente y desconocida.
- Cada diez años ocurría este crecimiento:

atributo	cambio
cantidad en uso	x100
velocidad de procesamiento	x100
capacidad de almacenamiento	x100
precio	/10

Los lenguajes de programación

- Son todos esencialmente equivalentes.
- Tienen una sintaxis y una semántica precisa.
- Hay lenguajes especializados para diferentes actividades de programación.
- Son estructuras lógicas precisas.
- Poseen una estructura léxica precisa.
- Son más estrictos que los lenguajes naturales.

Un ejemplo de programa

- Calcular el factorial M! = 1x2 x... xM, un ejemplo matemático clásico.
- Podemos escribir un programa simple que lo calcule.
- Está escrito en un lenguaje inexistente, pero parecido a todos los lenguajes reales.

Un programa simple

```
function Factorial(M)
   R=1
   for I=1 to M
        R = R*I
   end for
return R
```

valor inicial del resultado

ciclo de iteración

multiplicar por el valor siguiente

valor final del resultado

La numerabilidad de los conjuntos infinitos

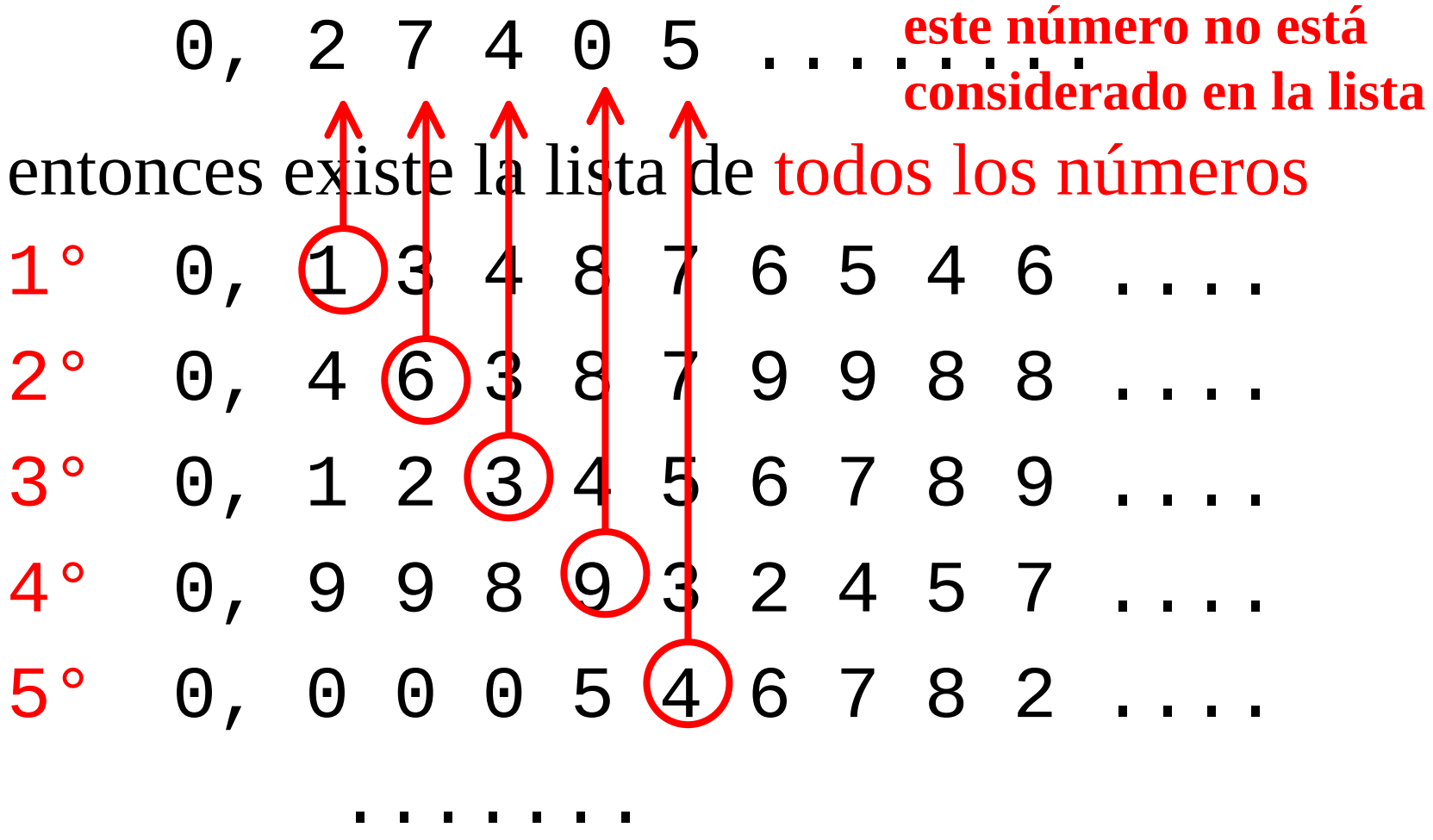
La noción de numerabilidad

- Numerar es poner en correspondencia con los números naturales. Es el concepto cotidiano.
- Los conjuntos finitos son numerables.
- La dificultad está en los conjuntos infinitos.
- Los números naturales son numerables.
- Los números enteros son doblemente infinitos pero también son numerables:
- Este procedimiento puede extenderse para más de dos series.

¿Hay conjuntos no numerables?

- Los números reales no son numerables. Esta es una propiedad básica del continuo.
- La demostración es simple y posee interés.
- Es un resultado de Georg Cantor (1845, 1918).
- Consideremos los números reales comprendidos entre 0 y 1, para simplificar.
- Si éstos no son numerables, tampoco lo será la totalidad de los números reales.

Supongamos que los reales fueran numerables

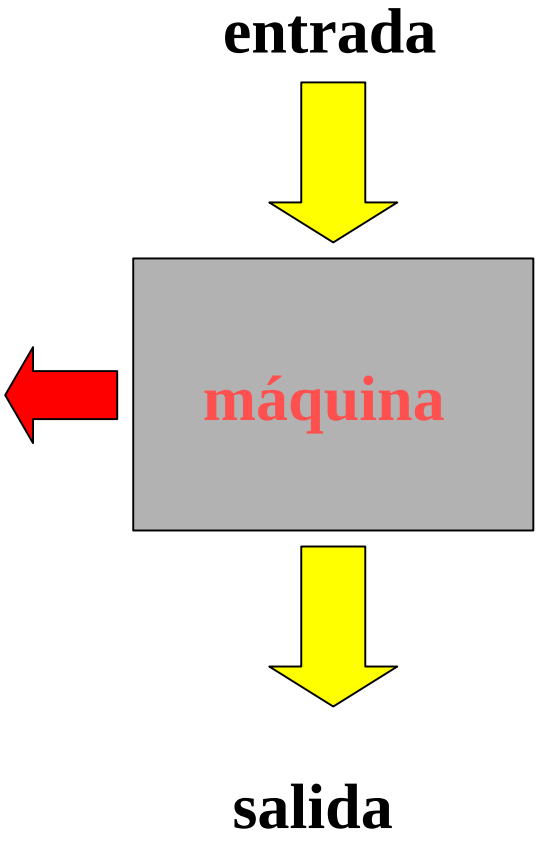


El resultado es general

- Una sucesión infinita de sucesiones infinitas tiene esta propiedad.
- La demostración se basa en:
  - la aplicación del "método diagonal"
  - una reducción al absurdo
- Estas son estrategias generales para la demostración de este tipo de resultados.

Turing y sus resultados

La idea de autómata o máquina finita



- La máquina tiene una entrada y una salida.
- En la entrada hay símbolos de un alfabeto finito.
- A la salida también hay símbolos del alfabeto finito.
- La máquina está formada por un número finito de “partes”.
- También podría existir un “efecto lateral”.

Limitaciones de la máquina finitas

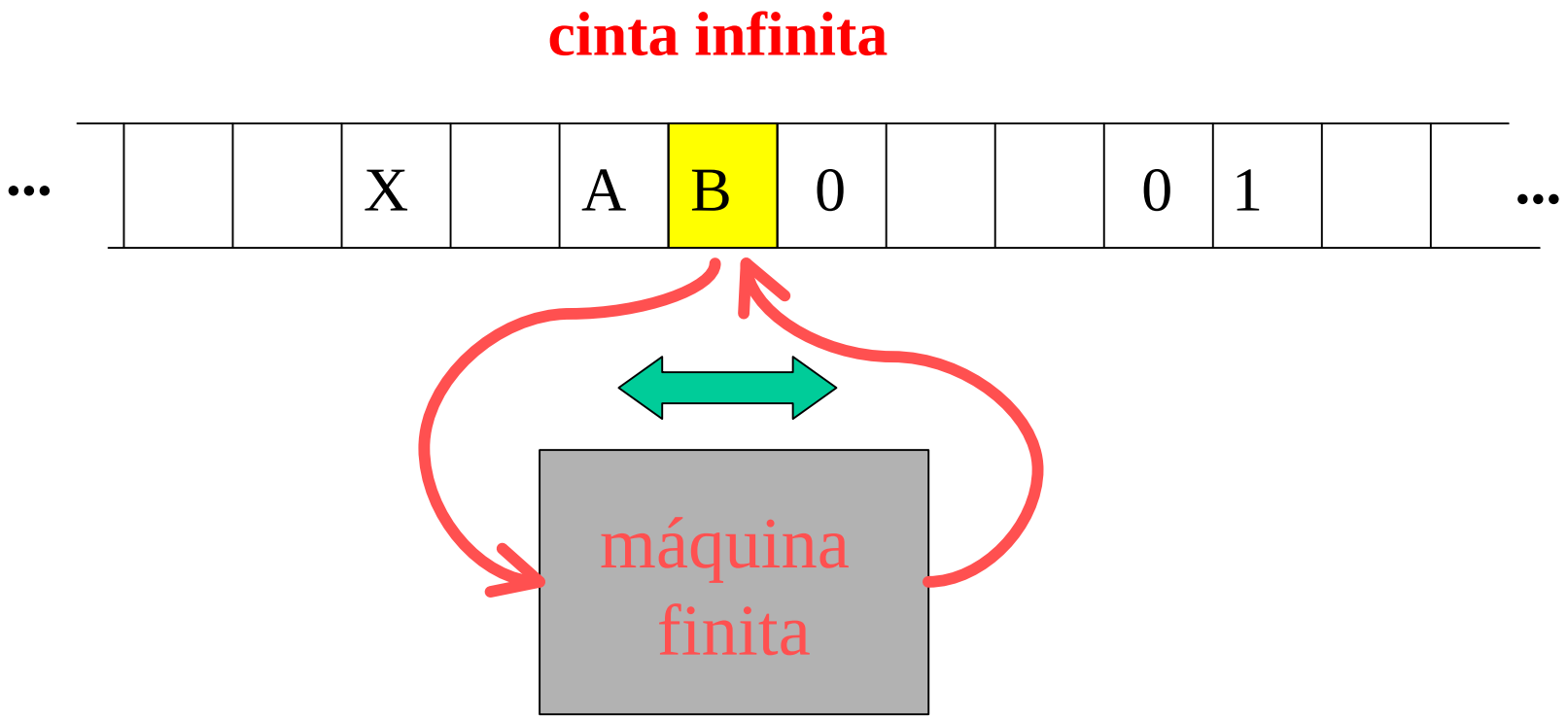
- Hay procesamientos que una máquina finita no puede hacer:
- Consideremos esta secuencia:

  0 1 0 1 1 0 1 1 1 0 1 1 1 1 0 1 1 1 1 1 0 ...
- Ninguna máquina finita puede generar esta secuencia.
- Tampoco puede investigar si una serie de paréntesis está “bien formada”.

Las máquinas infinitas

- Son necesarias aún para procesamientos simples, como verificar un sistema de paréntesis.
- La idea de Turing fue definir una máquina infinita simple, pero muy poderosa.
- Consta de dos elementos:
  - una máquina finita
  - una "cinta", doblemente infinita, donde se escriben y leen símbolos de un alfabeto finito

Los elementos de la máquina de Turing



En palabras

- La máquina trabaja en un lugar de la cinta.
- Su entrada es el símbolo que está escrito.
- Su salida es la salida de la máquina finita y se escribe en la cinta.
- El lugar de trabajo se desplaza una posición, o bien hacia la izquierda o bien hacia la derecha, según esté indique la máquina.
- La máquina eventualmente se detiene.

La máquina de Turing universal

- Es posible demostrar que existe una máquina de Turing capaz de simular la acción de cualquier otra máquina.
- Se divide la cinta en dos mitades.
- En la mitad izquierda se coloca una "copia" de la cinta de la máquina a simular.
- En la mitad derecha se colocan una "descripción" de la máquina a simular.

El autómata universal

- Consideremos ahora una máquina cualquiera, cualquiera sea su tecnología o su complejidad.
- Su acción puede ser simulada por la máquina de Turing universal.
- De aquí resulta la definición general de un autómata: es la actividad de una máquina de Turing.
- Las computadoras son máquinas universales (excepto por su memoria finita).

¿Por qué nos invaden las computadoras?

- Porque son autómatas universales.
- Pueden actuar como cualquier máquina (si se agregan sus efectos laterales).
- Tarde o temprano reemplazarán a todas las máquinas automáticas.
- Es un problema de velocidad de cálculo o de tamaño de memoria, nada más.

Variantes de la máquina universal

- Si un lenguaje de computadora permite construir una máquina de Turing universal, entonces es un lenguaje universal.
- En forma equivalente, un autómata es un programa escrito en un lenguaje universal.
- Autómata, programa y algoritmo son conceptos equivalentes.

La informática ayuda a la matemática

Los problemas de existencia

- En muchos casos en matemática el resultado que se busca es probar la existencia de un determinado objeto.
- Para este fin no importa el método que se siga, lo que interesa es el resultado.
- Un programa de computadora puede buscar objetos y ayudar a probar la existencia.
- Un matemático verifica que el objeto es correcto.

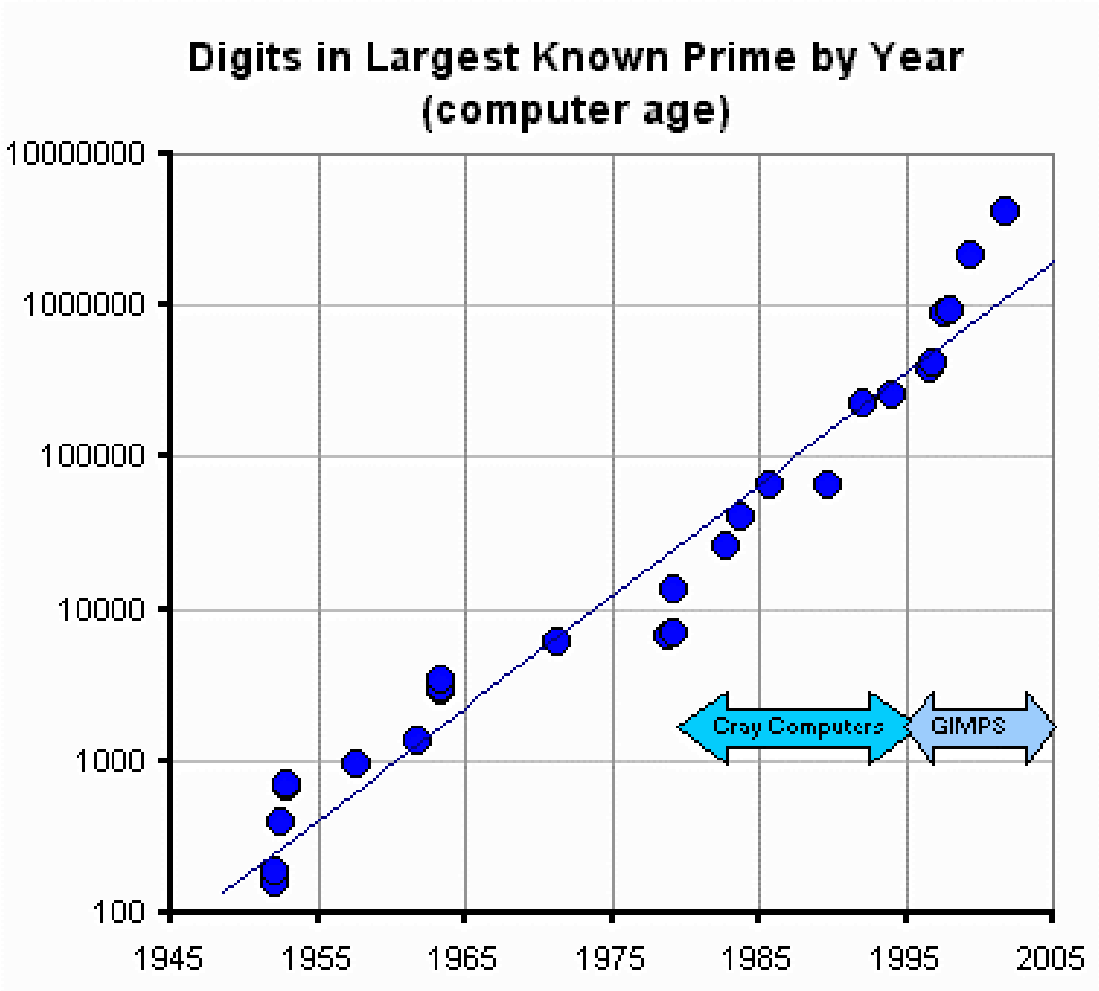
Algunas conclusiones

- En los problemas de existencia se pueden combinar programas de computadora con teoremas auxiliares.
- Los resultados son matemáticamente indudables si los resultados se pueden verificar a mano.
- Otro asunto es si no se pueden verificar a mano y hay que "demostrar" el programa.

La complejidad de los algoritmos

- En general, la complejidad para aplicar un algoritmo al número n se mide por la función f(n) vinculada a la cantidad de operaciones a realizar.
- Hay problemas polinómicos, como investigar si un número es primo.
- También hay problemas exponenciales.
- Consideremos la búsqueda de números primos

Resultados históricos



La evolución en el tiempo

- El número de dígitos del número primo n es proporcional a log n.
- Este número crece en forma lineal en el tiempo, con la ecuación aproximada:

  x20.000 en 60 años = x5,2 por década
- Este número se debe comparar con la velocidad de cálculo: x100 por década.
- La ecuación de complejidad sería $\sim(\log n)^{19}$.

Complejidad actual de los números primos

- Un teorema reciente (2002) establece que la complejidad de un nuevo método para determinar si un número n es primo es del orden $(\log n)^{12}$.
- Este resultado aumenta la velocidad histórica de los máximos números primos conocidos.
- Debería aumentar esta velocidad con los nuevos algoritmos.

La demostración de teoremas

- Es la propuesta original de Hilbert.
- La respuesta es negativa. No hay un algoritmo que permita decir si un teorema es correcto.
- Lo interesante es que hay otros resultado negativos también originados en las propuestas de Hilbert.
- Todo parece indicar que seguirán apareciendo resultados negativos sobre otras pruebas automáticas.

El décimo problema de Hilbert

- Hilbert propuso, también, otro caso de cálculo automático: un algoritmo que decidiera si una ecuación diofántica tiene solución.
- En 1970 el problema fue resuelto por Matyasevich y la respuesta es negativa.
- La prueba consistió en construir una ecuación diofántica cuya solución se vincula con números no computables mediante una máquina de Turing.

El problema de los 4 colores

- En 1852 Francis Guthrie preguntó si era posible demostrar la conjetura de que todo mapa se puede pintar con 4 colores.
- Este problema fue resuelto un siglo después.
- Hay diversos resultados que permiten reducir el problema general a casos especiales.
- En 1950 se lo había reducido a unos 10.000 casos.

... continuación

- En 1970 una nueva estrategia lo llevó a 1.000 casos.
- El problema era que una computadora de este momento emplearía unas 100.000 horas en analizarlos.
- En 1976 se logró analizar unos 2.000 casos, con una nueva estrategia: 1.200 horas.
- Es necesario probar que el programa es correcto desde el punto de vista lógico.

Los nuevos teoremas matemáticos

- Ha aparecido un nuevo tipo de teorema matemático.
- Su estructura es del tipo:

  demostración convencional

  programa de computadora

  demostración convencional
- Esto plantea nuevos desafíos metodológicos: demostrar programas.

La demostración de programas

- Es un problema lógico posible.
- Para que la demostración sea simple es conveniente que el programa esté bien estructurado.
- Esto ocurre cuando se emplean estructuras que poseen solamente un único punto de entrada y un único punto de finalización
- Un ejemplo práctico.

¿Este programa es correcto?

- Regresemos al problema de calcular el factorial M.
- ¿Cómo sabemos que el programa es correcto?

```
function Factorial(M)
    R=1
    for I=1 to M
        R = R*I
    end for
return R
```

¿qué pasa si M es cero?

¿qué pasa si M es negativo?

¿qué pasa si M no es entero?

... continuación

- La dificultad de la prueba lógica crece con el largo del programa.
- La prueba se vincula con la existencia de errores de programa.
- No hay ningún método algorítmico para verificar si un programa es correcto (puesto que no hay manera de demostrar teoremas automáticamente).

Los fenómenos aleatorios

- Son una parte importante de la ciencia y la metodología contemporánea.
- Nos concentraremos en un único aspecto: las secuencias aleatorias.
- El dilema todavía (creo que) no está resuelto desde el punto de vista conceptual.
- La definición ha cambiado en el tiempo.

Posibles definiciones

- Definiciones funcionales clásicas mediante tests.
- Definiciones axiomáticas.
- Definiciones “modernas” mediante la teoría de la complejidad.
- Solamente nos ocuparemos de éstas últimas, porque se vinculan con la informática.

Las nuevas ideas de aleatorio

- El único cambio significativo ocurrió con Kolmogorov, Chaitin y Solomonoff en los 60s.
- Estos autores introdujeron la idea de complejidad de una secuencia de símbolos:
- La complejidad K de una secuencia s es el largo mínimo de un programa (de una máquina universal), medido en bits, que genera esta secuencia.

Nueva definición de lo aleatorio

- La complejidad permite dar una definición nueva y muy fuerte de secuencia aleatoria.
- Sea una secuencia de bits de largo L, es una secuencia aleatoria si su complejidad es del orden de L.
- No aparece aquí ningún elemento dudoso o no definible.

Pertinencia de la definición

- Si una secuencia posee una regla simple de formación, no es aleatoria.
- Su complejidad es grande en la medida que es “caótica”.
- Un generador de números “aleatorios” posee un algoritmo de generación muy simple, luego está muy lejos de generar una secuencia aleatoria en este sentido.

Las limitaciones de la informática

Las limitaciones de los autómatas

- Los autómatas tienen dos tipos de limitaciones:
  - las limitaciones tecnológicas debido a que la memoria real no es infinita o debido al tiempo de procesamiento
  - las limitaciones teóricas
- Nos ocuparemos de las limitaciones teóricas.
- Estas limitaciones no desaparecerán en el futuro.

El problema de Halting

- Hay un problema de particular interés acerca de las máquinas de Turing (o de los programas de computadora). ¿Se detiene su marcha?
- Se trata de investigar si existe una manera automática de investigar si una máquina cualquiera finalmente se detiene o no.
- Este problema se llama problema de halting.

Demostración

- Consideremos un lenguaje universal de computación.
- Todos los programas que se pueden escribir se pueden numerar porque la cantidad de programas que se pueden escribir con N caracteres es finita.
- Sea n el programa enésimo de una numeración.
- Supongamos que existe una función Halting(n), escrita en este lenguaje de computación que cumple:
  - si el programa n se detiene, Halting(n) = 1
  - si el programa n no se detiene Halting(n) = 0
- La existencia de esta función nos llevará a una contradicción.

Consideremos una nueva función

```
function Turing(n)
UNO: if Halting(n) = 1 then
                goto UNO
           else return 1
end funtion
```

- Este programa se detiene si el programa n no se detiene (es decir, si Halting(n) = 0) y no se detiene si el programa n sí lo hace.

... continuación

- El programa anterior posee un número en la ordenación general de programas. Sea p este número.
- ¿Qué sucede cuando calculamos Turing(p)?
- La función Turing(p) se detiene y devuelve 1 si Halting(p) = 0 o sea si el programa p (la función Turing) no se detiene. Recíprocamente, no se detiene si el programa p sí lo hace.

... continuación

- La existencia de un algoritmo general (o un programa) que investigue si un programa se detiene nos conduce a una contradicción.
- Luego, no existe tal algoritmo.
- Este es uno de los resultados mayores de la informática.
- El teorema de Halting conduce a otros resultados de imposibilidad.

Los números computables

- Un número (real) se llama computable si existe un programa que es capaz de calcular los n primeros dígitos de su desarrollo decimal y luego detenerse.
- Los números racionales son computables.
- Hay números no computables.
- Esta una conclusión importante del teorema de Halting.

Los números *raíz* de 2, *pi* o *raíz* de *pi* son computables

- Hay fórmulas de recurrencia para todos estos casos. Luego, hay algoritmos y programas.
- De hecho todos los números que emplea efectivamente la matemática de las ciencias exactas y naturales son computables.
- La existencia de tablas numéricas anteriores a las computadoras es una prueba empírica de esta afirmación.

Existencia de números no computables.

- El conjunto de los programas de un lenguaje universal se pude numerar.
- El conjunto de los números reales no se puede numerar (teorema de Cantor).
- Luego, hay una infinidad de números que ni ahora ni nunca, por ninguna computadora, serán computables.

Una demostración constructiva

- Esta demostración no construye un número no computable, pero es sencillo construir uno.
- Sea el número cuyo desarrollo decimal es:

  0, a b c d e f ... p ...

  donde la cifra p vale 0 si el programa número p no se detiene y 1 si se detiene.
- Este número no es computable.

¿Una nueva matemática?

Las leyes computables

- Hasta el presente todas las leyes físicas conducen a resultados computables.
- Tal vez sean una excepción las leyes estadísticas o probabilísticas. Estas leyes son computables, pero parecen ser otro tipo nuevo de leyes.
- ¿Qué podemos concluir de este hecho?

... continuación

- Una conjetura posible es que se trata de una gigantesca casualidad. Con esta conjetura no se puede avanzar mucho.
- Una segunda conjetura es que todavía no hemos encontrado una ley no computable, pero que puede ocurrir, basta esperar.
- Una tercera conjetura es que todas las leyes del Universo son computables.

El Universo computable

- De las tres conjeturas, la única que permite extraer conclusiones de interés es la tercera.
- Si las leyes son todas computables, ¿por qué ocurre esto?
- Una respuesta posible es: porque el Universo no es continuo, sino numerable.
- Por esta arzón ocurre que todas las leyes físicas son computables.

Las cortaduras de Dedekind

- ¿Qué sucede entonces con la matemática?
- Regresemos a Dedekind y la cortadura:
  - Es una clasificación de los números racionales en dos clases, no vacías.
  - Todo número racional está clasificado y pertenece solamente a una de las clases.
  - Todo elemento de la clase inferior **I** es inferior a todo elemento de la clase superior **S**.
- ¿Está bien esta definición?

La omisión de Dedekind

- La cortadura exige que haya un método tal que dado un racional r, se sepa a qué clase pertenece.
- Este método debe ser preciso e inequívoco.
- En otras palabras, debe existir un algoritmo que dado r me devuelva I o S.
- Si esto es así, la cortadura define un número computable y no un número real.

¿Entonces?

- Yo estoy convencido que el continuo es una exageración de los matemáticos.
- Los números reales no computables no se pueden usar ni conocer. No sirven para nada.
- Dedekind no sospechaba las dificultades de su definición porque faltaban muchas décadas para que se precisaran estas ideas.

¿Es continua la geometría?

- El mismo problema se plantea en la geometría.
- Puede pensarse que es necesaria la continuidad la geometría, pero esto no es así.
- Es posible una geometría solamente de elementos computables.
- Es la geometría analítica donde todas las ecuaciones son funciones computables.

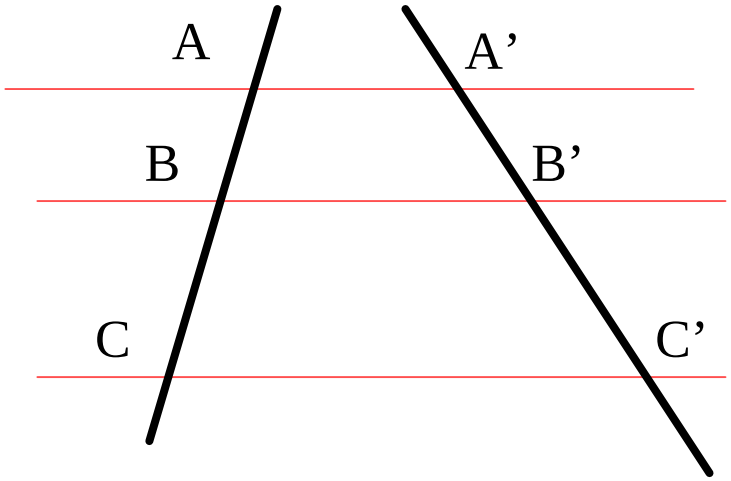
... continuación

- Las rectas se cortan, cortan a los círculos, forman triángulos. Las ecuaciones de la geometría analítica así lo demuestran.
- No hay manera de diferenciar la geometría computable de la geometría continua.
- Podemos construir objetos geométricos no computables, pero solamente con axiomas que así lo permitan expresamente.

El teorema de Thales

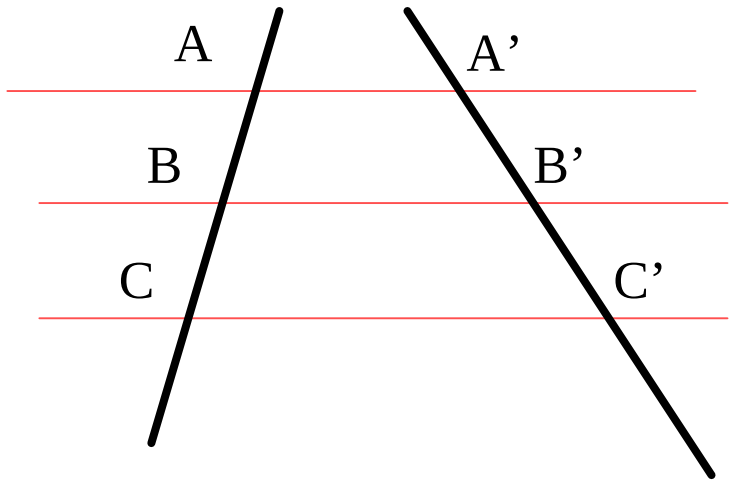
- Es un teorema simple de la geometría, pero que no es fácil de demostrar.
- Requiere los axiomas del continuo.
- Por esta razón es interesante analizarlo como ejemplo de una “nueva” geometría que no emplea la noción de continuo.
- Sería el resultado de aplicar la navaja de Occam a la geometría.

... continuación



- El teorema dice que: AB/BC = A’B’/B’C’
- El continuo es necesario porque la razón de los segmentos podría ser irracional, por ejemplo.

... continuación



- El teorema es trivial si: AB = BC
- Como consecuencia, si la razón es racional, se reduce a aplicar varias veces el caso anterior.

... continuación

- El caso racional se pude generalizar al caso de razones computables.
- Si la razón AB/BC es computable, entonces también la razón A'B'/B'C' será computable, basta aplicar el mismo algoritmo a la otra serie de segmentos.
- Por extensión del caso racional, vale el teorema de Thales para las razones computables.

Conclusiones

- La introducción de la continuidad aparece como una forma de “abuso” no necesario.
- Creo que se ganaría mucha claridad en la matemática si se reemplazara la noción de número real por la noción de número computable.
- La matemática sería muy poco diferente. Casi no existirían cambios de enunciados.

Esta es una idea personal, solamente, no una verdad aceptada

¡Muchas gracias por su paciencia!