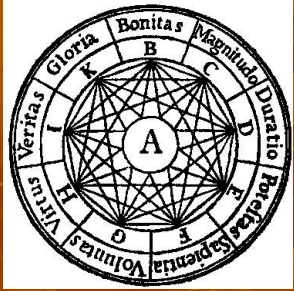


Ars Magna de Ramón Lull

(1232 – 1315)



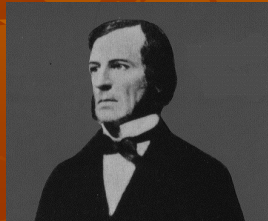
Máquinas calculadoras de Pascal y Leibniz, Sistema Binario



Leibniz (1646 – 1716)

Siglo XIX

Lógicas booleanas/
álgebras booleanas/
circuitos booleanos.

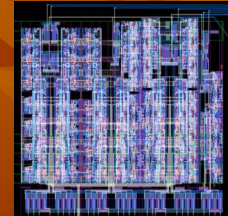
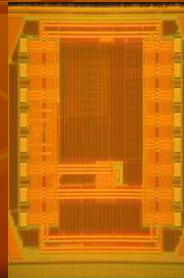


Boole (1815 - 1864)

Piano lógico de Jevons.

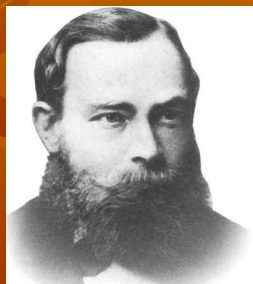
Circuitos lógicos (cálculo proposicional)

Flip-Flop hecho por
Kwangil Oh in 2003.



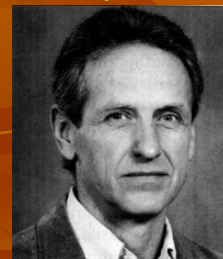
500MHz Delay-
locked loop (DLL)
fabricado por Byung-
Guk Kim, 2004.

Primer sistema
completo del cálculo de
relaciones, en 1879.



Frege (1848 – 1925)

Programación lógica (negación como falla, resolución y unificación)



Kowalski (1941 --)

Lógica experimental de Annibale Pastore



Pastore (1868 – 1956)

Auxilios para pensar



Teoría de funciones recursivas

Turing (1912 – 1954)



Programación funcional (cálculo lambda).

Church (1903 – 1995)

Modelo computacional por reescritura de cadenas



Post (1897 – 1954)

Lógicas temporales

- Semánticas para programas (lógicas lambda, temporales, dinámicas y lineales),
- Control de sistemas (temporales, dinámicas),



Prior (1914 – 1969)

Lógicas epistémicas, programación en paralelo, concurrencia de multi-agentes



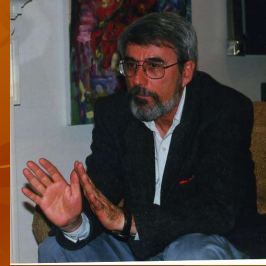
Hintikka (1929-)

Lógicas borrosas o difusas (fuzzy)



Zadeh (1921-)

Lógicas no-monotónicas: Las conclusiones son retractables a la luz de información adicional



Reiter (1939 – 2002)

Un cambio conceptual que todavía está en proceso

- En el “Modelo Antiguo” empezábamos con axiomas intuitivos y procedíamos mediante deducción.
- Esto hacía innecesario investigar a fondo el manejo del error o la revisión.
- Después de todo, los sistemas deductivos clásicos tienen automáticamente monotonicidad por ser compactos.

Pero el mundo cambia y los agentes crean.

- ¡No son errores lógicos ni epistémicos!
- La información requiere tanto recolección (que puede ser equivocada, contradictoria, incompleta) como procesamiento (que puede saltar a conclusiones para no ser demasiado lento).

Tipos de razonamientos no deductivos

- Inductivos: Se generaliza a partir de algunos casos
- Abductivos: Se concluye una explicación
- Probables: Frecuentista o bayesiano
- Estadísticos: Por muestreo
- Prima facie: A falta de información en contra

- Inciertos: Con reglas o premisas falibles
- Plausibles: Altamente apoyados por la evidencia
- Retractable: Bloqueables si el contexto cambia
- Default: Tipicamente
- De sentido común: Aceptable para una comunidad

Hay que permitir la modelación de la no-monotonicidad.

- Sin esto no hay un manejo adecuado de los bancos de datos “deductivos” que frecuentemente asumen el *supuesto de mundo cerrado* (“closed world assumption”) y utilizan varios tipos de circunscripción (especialmente de dominio y de predicado).
- En robótica, la no-monotonicidad auxilia con el problema del marco de coordenadas (frame) que indican qué propiedades o dimensiones cambian mientras otras permanecen.

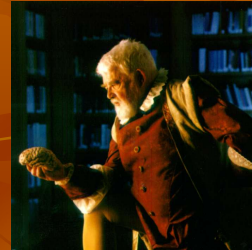
Necesidad de no-monotonicidad

- En el diseño de sistemas, proporciona mayor tolerancia a fallas; algo muy importante pues la complejidad de los sistemas interesantes lleva a la fragilidad.
- También se aplica al reconocimiento de patrones ya que es común que hagamos conjeturas visuales o fonológicas retractables.

El tratamiento lógico de la no-monotonicidad nos ayuda a manejar

- la programación lógica con su noción de negación como falla,
- la teoría de los sistemas expertos y sus grados de confiabilidad,
- la representación del conocimiento mediante prototipos y herencia (programación por objetos OOP), y
- la búsqueda en inteligencia artificial del elusivo sentido común.

Inteligencia artificial: Sentido común por circunscripción



McCarthy (1927-)

En resumen

- Además de todas sus contribuciones clásicas, las nuevas lógicas y los nuevos desarrollos de otras prometen ayudarnos a incorporar y modelar computacionalmente una amplia gama de razonamientos no deductivos.



BROUWER, LUKASIEWICZ, RUSSELL, GENTZEN, TARSKI, GÖDEL

- La larga lista de contribuciones de la lógica al desarrollo, sistematización y entendimiento de la computación, promete ser incrementada exponencialmente en el siglo XXI. Sería una lástima que nos perdiéramos de estos beneficios por un incompleto conocimiento de ellos.