El Programa de Hilbert.

En un principio, se dio por supuesto que todo problema enunciado en términos matemáticos podría resolverse. Para Hilbert era esencial que todo problema tuviera una solución exacta. Surgieron algunas paradojas que hicieron que Hilbert se plantease un **programa formalista** (trataba de establecer axiomas y reglas irrefutables). Su idea era obtener respuesta a:

- ¿Son las matemáticas **completas** (cualquier proposición puede ser probada o rechazada)?
- ¿Son **consistentes** (no es posible demostrar algo falso)?
- ¿Son **decidibles** (se puede determinar de forma mecánica si una proposición podrá demostrarse como cierta o falsa tras una secuencia finita de pasos? (Entscheidugnsproblem)

Hilbert creía que la respuesta era afirmativa para las tres.

Kurt Gödel. (Para las dos primeras preguntas).

Demostró, mediante los Teoremas de Incompletitud, que las dos primeras preguntas no son ciertas para una formalización que contenga la Aritmética de Peano.

- **Primer Teorema de Gödel:**
  Cualquier sistema formal que contenga la aritmética de Peano, o es incompleto o es inconsistente.
- **Segundo Teorema de Gödel:**
  Ningún sistema consistente se puede usar para demostrar su propia consistencia.

La demostración del primer teorema de Gödel (NPR* NPR*) nos da como consecuencia que cualquier máquina que produzca declaraciones sobre la aritmética a veces produce declaraciones falsas, o de lo contrario hay afirmaciones verdaderas acerca de la aritmética que nunca produce. Esto supone una tragedia para el problema de Hilbert, pues si queremos saber la verdad, y nada más que la verdad, no podremos saber toda la verdad. La verdad absoluta no existe ni en las matemáticas.

Enscheidugnsproblem(problema de decisión) (tercera pregunta): Como concepto de procedimiento mecánico disponemos de la Máquina de Turing.

- La máquina de Turing (qué es):
  Turing trato de modelar una computadora humana:
  - Se escriben símbolos de un conjunto finito.
  - Cada acción depende del símbolo que es examinado y del "estado mental" del momento.
  - Los estados mentales pueden cambiar pero son finitos

  Una **Máquina de Turing** es un dispositivo dotado de una cinta de memoria ilimitada, formada por celdas donde se pueden leer y escribir símbolos. En cualquier instante sólo hay un número finito de celdas en uso, quedando las demás en blanco. En dicha cinta existe un cabezal que puede estar en diferentes estados (representan los estados mentales).

  Disponemos de un alfabeto de entrada y otro de los símbolos usados en la máquina.

- La máquina de Turing (cómo funciona):
  - En cada paso, el cabezal lee la posición en la que se encuentra en la cinta, y dependiendo del estado, escribe; posteriormente avanza o retrocede a la celda adyacente.
  - Puede cambiar de estado.
  - Repetimos hasta que no tenga definida una tarea

Son una buena formalización del concepto de algoritmo porque:

1. Cada programa de un MT puede ser implementado.
2. Todos los algoritmos conocidos han podido ser implementados en MT.
3. Cualquier otro intento de formalizar dicho concepto fueron reducidos o equivalentes a la MT.

Usando la MT se demuestra que la tercera pregunta no es cierta. Destruye el programa de Hilbert pero da lugar a la Teoría de la Computación.

El problema de la parada (Halting Problem):

Es una versión del problema de decisión para el cual no hay respuesta en un número finito de pasos.

Se trata de diseñar un algoritmo con entrada un programa y unos datos. Como salida obtenemos sí (si el programa para para esos datos) o no (cuando queda ciclando indefinidamente).

Termina (P,x) {True si P(x) termina, False en otro caso}

Complejidad-computacional.

Pretendemos medir la complejidad computacional de un problema. Dado un problema P, decimos que es de complejidad temporal **O(T(n))** si existe un algoritmo que lo resuelve en a lo más cT(n) instrucciones elementales sobre una entrada de longitud n.

Los problemas **tratables** son aquellos para los que existen algoritmos polinómicos e **intratables** el resto.

El concepto de problema.

Problemas de optimización: tratan de maximizar o minimizar una cantidad.

Problemas de decisión: la solución solo puede ser SI o NO.
Disponemos de: Alfabeto A: conjunto finito de símbolos.

Palabra w: secuencia finita de símbolos de A.
A*: Conjunto de todas las palabras construidas posibles.
Lenguaje L: Conjunto de palabras.
Problema de decisión asociado a un lenguaje L: Dado $w \in A^*$, decidir si $w \in L$. Los problemas de cálculo se pueden transformar en problemas de este tipo.
La complejidad de un lenguaje L es la complejidad del problema de decisión asociado a L. L puede ser solucionado eficientemente cuando existe un algoritmo polinómico que decide L. En otro caso se dice que L es un problema difícil.