Embelleciendo a Gödel

Eric C.R. Hehner - Universidad de Toronto

1990

1. Introducción

Los teoremas de la incompletitud de Kurt Gödel [1931] se consideran entre los más importantes resultados de la matemática. Son considerados "profundos", y han influenciado fuertemente el curso de la matemática moderna. Se cree popularmente que prueban los límites (¡o incluso la inutilidad!) del formalismo matemático. En cualquier caso, merecen ser presentados lo más simple y con la mayor elegancia posible. La presentación original de Gödel fue cuidadosa y clara, pero no tan simple como podría ser.

Nuestro embellecimiento se realiza en etapas. Es tentador saltearse las etapas y presentar sólo la forma final: una prueba de tres líneas. Desafortunadamente las etapas son necesarias para convencer al lector de que la esencia de los teoremas de Gödel no se ha perdido. Y hay algunas lecciones que aprender en el camino. Los teoremas pertenecen a la rama de las matemáticas conocida como metamatemática, por lo que empezaremos por ahí.

2. Metamatemáticas: El Estudio de los Formalismos

Para el estudio de las estrellas, es útil diseñar un formalismo matemático para ello. Sin embargo, la matemática no esta limitada al estudio de la naturaleza. También nos ayuda a estudiar los mundos artificiales de los puentes, la economía, la música, e incluso los procesos del pensamiento. Y si queremos que los formalismos matemáticos sean objetos de estudio, es útil diseñar un formalismo matemático para ello.

Diseñamos un formalismo (sinónimo: una teoría) para que sus sentencias constituyan declaraciones sobre los objetos de estudio. Algunas de estas declaraciones serán verdaderas y otras falsas, diseñamos la teoría a fin de que sus teoremas (sentencias demostrables) representen declaraciones verdaderas y sus antiteoremas (sentencias refutables) representen declaraciones falsas. Supongamos, por ejemplo, que nuestro objeto de estudio es una versión de la teoría de números, que llamaremos TN.

Traducción del inglés por Pablo G. Ventura - Enero de 2012

He aquí ocho ejemplos de afirmaciones verdaderas acerca de (pero no en) TN:

(a) $1 + 1 = 2$ es un teorema de TN.

(b) $1 + 1 = 3$ es un antiteorema de TN.

(c) $0 \div 0 = 4$ no es un teorema ni un antiteorema de TN.

(d) $(0 \div 0 = 4) \vee \neg(0 \div 0 = 4)$ es un teorema de TN.

(e) Una sentencia es un antiteorema de TN si y sólo si es la negación de un teorema de TN.

(f) Una sentencia es un teorema de TN si y sólo si su negación es un antiteorema de TN.

(g) Ninguna sentencia es a la vez un teorema y un antiteorema de TN.

(h) Toda sentencia de la forma $s \vee \neg s$ es un teorema de TN.

He aquí seis declaraciones falsas sobre TN:

(i) $1 + 1 = 2$ es un antiteorema de TN.

(j) $1 + 1 = 3$ es un teorema de TN.

(k) $0 \div 0 = 4$ es siempre un teorema o un antiteorema de TN.

(l) Una sentencia es un antiteorema de TN si y sólo si no es un teorema de TN.

(m) Una sentencia es siempre un teorema o un antiteorema de TN.

(n) Siempre una sentencia o su negación es un teorema de TN.

La declaración (a) nos muestra un teorema simple. En los textos de matemáticas, se ahorra espacio para escribir una sentencia como $1 + 1 = 2$ sin decir nada al respecto, lo que significa que es un teorema. Pero en este trabajo no lo haremos. La declaración (g) dice que TN es consistente. La declaración falsa (m) dice que TN es completa. Incluimos la división en TN para dar una sentencia simple que no es ni un teorema, ni un antiteorema. El resultado sorpresivo de Gödel fue que incluso sin división, con sólo la adición y la multiplicación, hay sentencias que no son ni teoremas ni antiteoremas.

Vamos a llamar a nuestra teoría para estudiar las teorías TT. Para cualquier teoría T y una sentencia s de T introduciremos la sentencia (de TT):

$$T \vdash s$$

para representar la declaración (ya sea verdadera o falsa) de que s es un teorema de T. E introduciremos

$$T \dashv s$$

para representar que la declaración de que s es un antiteorema de T. Cuando es evidente la teoría que está en estudio, podemos omitir su nombre y escribir simplemente $\vdash s$ y $\dashv s$. Debido a la declaración (e) podemos tener la tentación de prescindir del símbolo de "antiteorema" y en vez, hablar de la negación de un teorema. Sin embargo, no es necesario que todas las teorías incluyan la negación, y puede ser interesante estudiar algunas que no la incluyen. Debido a (c), ciertamente no podemos tomar "es un antiteorema" como "no es un teorema". Necesitamos un símbolo de "antiteorema" por la misma razón que el álgebra de Boole necesita de ambos símbolos para "verdadero" y "falso", de hecho "verdadero" y "falso" son un teorema primitivo y un antiteorema respectivamente. También necesitaremos símbolos para el "o", el "y", la "negación", el "si y sólo si", y los cuantificadores, como se ve en las declaraciones de nuestro ejemplo. Para evitar confusiones, podríamos insistir (como lo hizo Kleene) en que los símbolos de la teoría de TT difieren de los de las teorías en estudio. Sin embargo, es posible que desee utilizar TT para estudiar TT. Como veremos en el siguiente párrafo, hay una manera mejor de evitar confusiones. Así que reutilizaremos $\vee$, $\wedge$, $\neg$, $=$, $\forall$ y $\exists$ en TT.

La manera de evitar la confusión era conocida por Gödel y es bien conocida por los programadores: se trata de distinguir el programa de los datos. El programador de un compilador conoce la diferencia entre su programa y sus datos, a pesar de que sus datos son otro programa, aún si están en el mismo lenguaje. Para él, los datos de entrada son una cadena de caracteres, y su programa examina sus caracteres. Del mismo modo, en la metamatemática una teoría puede describir a otra sin confusión, incluso si esa otra teoría es ella misma, dándose cuenta de que, para describir la teoría, las expresiones de la teoría descripta son datos de tipo cadena de caracteres.

En la lógica moderna, la distinción entre el programa y los datos no siempre se hace, y $\vdash$ se aplica directamente a las sentencias. Para compensarlo parcialmente, los lógicos distinguen los operadores entre "extensionales" y "intensionales", y dictan reglas que indican cuando algo no puede ser sustituido por su equivalente. En aras de la simplicidad y claridad, vamos a mantener la distinción del programador: aplicamos $\vdash$ a una cadena de caracteres que representan una sentencia. Así

$$TN \vdash \text{“1+1=2”}$$

es la sentencia de TT que representa la declaración (a).

Si se omite el nombre de TN, los estados (a) a (n) están representados (formalmente) en TT de la siguiente manera. (Yuxtaposición de cadenas de caracteres indica concatenación, ¿la cuantificación esta sobre las cadenas de caracteres?)

Declaraciones verdaderas:

(aa) $\vdash$ "1+1=2"

(bb) $\dashv$ "1+1=3"

(cc) $\neg \vdash$ "0÷0=4" $\wedge \neg \dashv$ "0÷0=4"

(dd) $\vdash$ "(0÷0=4) $\vee$ $\neg$(0÷0=4)"

(ee) $\forall\, s,\ \dashv s = \vdash$ ("$\neg$"s)

(ff) $\forall\ s,\ \vdash s = \dashv (\text{“}\neg\text{”}s)$

(gg) $\neg\exists\ s,\ \vdash s \wedge \dashv s$

(hh) $\forall\ s,\ \vdash (s\text{“}\vee\neg\text{”}s)$

Declaraciones falsas:

(ii) $\dashv$ “1+1=2”

(jj) $\vdash$ “1+1=3”

(kk) $\vdash$ “0÷0=4”$\vee \dashv$ “0÷0=4”

(ll) $\forall\ s,\ \dashv s =\ \neg \vdash s$

(mm) $\forall\ s,\ \vdash s \vee \dashv s$

(nn) $\forall\ s,\ \vdash s \vee \vdash (\text{“}\neg\text{”}s)$

Tratamos de diseñar TT para que desde (a) hasta (h) sean teoremas y desde (i) a (n) sean antiteoremas. A la hora de diseñar una teoría para estudiar las estrellas, siempre debemos mantener algunas dudas sobre la capacidad de nuestra teoría de coincidir con los hechos. Lo mismo vale para una teoría para estudiar las teorías.

3. La matemática clásica y la constructiva

En la sección anterior, las declaraciones (h) y (n) son muy similares, pero (h) es verdadera y (n) falsa. ¿Cuál de estas es la Ley del Tercero Excluido? La verdad de esta “ley” ha suscitado intensos debates durante algún tiempo por los matemáticos que llevaron a cabo sus matemáticas informalmente; tal vez la informalidad era necesaria para la controversia. Para una teoría formal como TN, podemos distinguir (h), que es la “ley”, de (n), que (dado (e)) es una declaración de completitud.

Hay teorías interesantes donde la Ley del Tercero Excluido no vale, sino que para la prueba de una disyunción se requiere la prueba de una de las disyunciones. Podemos representar esto en TT de la siguiente manera.

$$\forall\ s, \forall\ t,\ \ \vdash (s\text{“}\vee\text{”}t) = \vdash s \vee \vdash t$$

Llamamos a estas teorías “constructivas”. Del mismo modo una prueba de la forma $\exists\ s, P(s)$ requiere un término t tal que $P(t)$ sea un teorema. Una prueba de la forma $\forall\ x, \exists\ y, P(x,y)$ es un programa que construye a partir de la entrada x, una salida y satisfactoria.

Las teorías en las que la ley del Tercero Excluido es válida, se denominan "clásicas". La mayoría de la gente está dispuesta a aceptar que "Dios existe o Dios no existe", sin una prueba de cualquier disyuntiva. Estas personas prefieren una teoría clásica en la que

$$(0 \div 0 = 4) \vee \neg(0 \div 0 = 4)$$

es un teorema aún sin que ninguna de las disyunciones sea un teorema. En la siguiente sección probaremos la igualdad de las dos sentencias, sin que sean ni un teorema, ni un antiteorema.

La discusión anterior oculta un punto sutil. Hemos dicho que en una teoría clásica, una disyunción puede ser un teorema, aunque ninguna de las disyunciones lo sea. Si nuestra metateoría TT es clásica, entonces tal vez $\vdash s \vee \vdash t$ sea un teorema (para algún s y t), aunque ninguna de las disyunciones lo sea. En ese caso la sentencia de TT

$$\forall\ s, \forall\ t,\ \vdash (s\text{“}\vee\text{”}t) = \vdash s \vee\ \vdash t$$

no representa nuestra intención de describir una teoría constructiva. Por esta razón, preferimos que nuestra metateoría sea constructiva. Desafortunadamente, si la metateoría se utiliza para describirse a sí misma, no nos puede decir si tiene esta propiedad constructiva.

4. El Primer Teorema de Incompletitud de Gödel

A grandes rasgos, el argumento de Gödel es el siguiente. él primero creó un elaborado plan para codificar sentencias como números. Luego codificó una sentencia que, en la superficie, se refiere a los números naturales, pero que puede ser vista como sinónimo de "yo no soy un teorema". El "yo" en esa sentencia es el número que codifica la sentencia. Es la Paradoja del Mentiroso en un traje nuevo. Si esta sentencia es un teorema, esta diciendo algo falso, por lo que la lógica es inconsistente. Suponiendo, como es razonable, que su lógica (*Principia Mathematica* de Russell y Whitehead) es consistente, se concluye que la sentencia es verdadera e indemostrable.

Gödel muestra cómo construir su sentencia verdadera pero indemostrable, pero no la construye. Esta construcción incluiría números inmensos, y la codificación es tan opaca que no hubíeramos podido ser iluminados por ella. Con el fin de examinar la sentencia de Gödel, deberíamos utilizar la codificación transparente de las secciones anteriores: las cadenas de caracteres.

Gödel definió el predicado "es un teorema" (lo llamó *Bew*) en los números (un predicado es una función de tipo booleano). Nosotros en su lugar definimos $\vdash$ para las cadenas de caracteres, de forma tal que $\vdash s$ es un teorema si y sólo si s representa un teorema. Esto se puede hacer; ya sea agregando el símbolo $\vdash$ al formalismo y dar las reglas para su uso, o mediante la definición de una función utilizando los símbolos que ya están en el formalismo. De cualquier manera, esto equivale a escribir un programa demostrador de teoremas. El punto importante es que tenemos una única teoría que sirve como objeto de estudio y como el formalismo con el que realizar el estudio.

A continuación, Gödel define una relación de encaje Q, una vez más sobre los números, nosotros deberíamos definir Q como una función de cadenas a cadenas. Cuando es aplicada a una cadena que representa un predicado, devuelve otra cadena que representa una sentencia mediante la sustitución de todas las ocurrencias de las subcadenas que representan variables libres, con la cadena completa entre comillas. Por ejemplo, para la cadena

$$\text{“}\exists\ u,\ t = u\ u\text{”}$$

representa un predicado en la variable libre t. Aplicando Q a esta cadena, obtenemos el teorema.

$$Q(\text{“}\exists\ u,\ t = u\ u\text{”}) = \text{“}\exists\ u,\ \underline{\text{“}}\exists\ u,\ t = u\ u\underline{\text{”}} = u\ u\text{”}$$

Las comillas internas están subrayadas para indicar que no son más que caracteres de la cadena. (Cada lenguaje de programación tiene una convención especial para escribir comillas adentro de las cadenas.) Una vez más, esta función se puede definir agregando el símbolo Q al formalismo y dando las reglas para su uso, o mediante la definición de una función utilizando los símbolos que ya están en el formalismo. De cualquier manera, equivale a escribir un programa.

Ahora todas las piezas están en su lugar. En particular, tenemos el teorema:

$$Q(\text{“}\neg \vdash Q(s)\text{”}) = \text{“}\neg \vdash Q(\underline{\text{“}}\neg \vdash Q(s)\underline{\text{”}})\text{”}$$

Aplicando $\neg \vdash$ a ambos miembros de la ecuación anterior, obtenemos el teorema:

$$\neg \vdash Q(\text{“}\neg \vdash Q(s)\text{”}) = \neg \vdash \text{“}\neg \vdash Q(\underline{\text{“}}\neg \vdash Q(s)\underline{\text{”}})\text{”}$$

Tomando con libertad la sustitución de las cadenas entre comillas, se puede decir que este teorema tiene la forma:

$$G = \neg \vdash \text{“}G\text{”}$$

donde cada ocurrencia de G representa $\neg \vdash Q(\text{“}\neg \vdash Q(s)\text{”})$.

Esta es la famosa sentencia de Gödel, pero utilizando como codificación las cadenas de caracteres. Si G fuera un teorema, entonces $\vdash$ “G” sería un teorema y $\neg \vdash$ “G”, un antiteorema. Así, podríamos probar la equivalencia entre un teorema y un antiteorema, por lo que la teoría sería inconsistente. Pero nosotros creemos que la teoría es consistente. De esta forma llegamos a la conclusión de que G no es un teorema. Si G fuera un antiteorema, entonces $\neg \vdash$ “G” debería ser un antiteorema porque se han demostrado iguales, por lo tanto $\vdash$ “G” debería ser un teorema. A partir de que $\vdash$ “G” es un teorema si y sólo si G es un teorema, G sería un teorema, lo que es inconsistente. Por lo tanto G tampoco es un antiteorema. La teoría es incompleta.

¿Cómo deberíamos interpretar G? Para que represente una declaración verdadera, por lo menos tiene que ser una sentencia formal. ¿Lo es? Para establecer que la declaración de Gödel

$$\neg \vdash Q(\text{“}\neg \vdash Q(s)\text{”})$$

es una sentencia, debemos mostrar que

$$\vdash Q(\text{“}\neg \vdash Q(s)\text{”})$$

es una sentencia. Si $\vdash$ se aplica sólo a cadenas que representan sentencias, entonces debemos mostrar que

$$Q(\text{“}\neg \vdash Q(s)\text{”})$$

representa una sentencia. Ya que podemos probar que

$$Q(\text{“}\neg \vdash Q(s)\text{”}) = \text{“}\neg \vdash Q(\underline{\text{“}}\neg \vdash Q(s)\underline{\text{”}})\text{”}$$

podemos preguntarnos si la cadena

$$\text{“}\neg \vdash Q(\underline{\text{“}}\neg \vdash Q(s)\underline{\text{”}})\text{”}$$

representa una sentencia. Esto sucede si y solo si

$$\neg \vdash Q(\text{“}\neg \vdash Q(s)\text{”})$$

que era la pregunta original. La única salida es permitir que $Q(s)$ sea una sentencia para cualquier cadena s, y/o que $\vdash s$ sea una sentencia para cualquier cadena s, aunque s represente o no una sentencia. Debemos, por lo menos temporalmente, suspender nuestro deseo de interpretar, debemos tener en cuenta sólo la manipulación formal de los símbolos con el fin de llegar al resultado de Gödel.

Teniendo en cuenta el axioma de la reflexividad de la igualdad, un formalista está dispuesto a aceptar $0 \div 0 = 0 \div 0$ como un teorema de TN. No tiene que ver con qué objeto matemático es mencionado por $0 \div 0$. Esta igualmente dispuesto a aceptar $\vdash$ “$) + x = ($” como una sentencia válida de TT, aunque no sea un teorema. Definimos $\vdash$ como predicado para las cadenas, de modo que $\vdash s$ es una sentencia sin importar qué cadena pueda ser s. Del mismo modo, Gödel, en su presentación formal, utiliza una codificación que asigna un número natural distinto a cada secuencia de símbolos, sea una expresión o no, y define su predicado *Bew* para todos los números naturales. Antes de la presentación formal de su resultado, Gödel da una presentación informal como motivación, en la que propone para codificar sólo los predicados en los números naturales: *“Pensamos los signos de clase [predicados] como si estuvieran dispuestos en una serie, y el n-ésimo es denotado por R(n)”*. Esto es posible porque el número del predicado es definido por un programa, este programa puede imprimir una lista de todos y sólo los predicados, asignando a cada uno el número natural siguiente. Esta codificación tiene el atractivo de que las sentencias y predicados metamatemáticos pueden siempre ser interpretados, a través de la codificación, como si se estuviera hablando de sentencias y predicados. Sin embargo, esta es una atracción fatal. De hecho, en la introducción a la traducción al inglés del trabajo de Gödel, R. B. Braithwaite considera como un defecto de la presentación formal, que Gödel haya codificado todas las secuencias de símbolos. Lejos de ser un defecto, ¡es esencial! Más tarde, estaremos en condiciones de hacer este punto más simple y claro.

5. El Segundo Teorema de Incompletitud de Gödel

El Primer Teorema de Incompletitud de Gödel dice que una teoría particular, si es consistente, es incompleta. Su interés proviene del esfuerzo que dedicó a tratar de hacer que la teoría fuese completa. Cuando se descubre que una sentencia no es ni un teorema ni antiteorema, puede ser uno o el otro, a nuestra elección, mediante la adición de un axioma. El Segundo Teorema de Incompletitud de Gödel dice que este proceso de adición de axiomas no puede hacer a la teoría completa (y seguir siendo consistente).

Cuando se añade un axioma a una teoría, se obtiene una teoría diferente. Para el segundo teorema tenemos que poner de nuevo el nombre de la teoría frente a $\vdash$ y $\dashv$. El primer teorema dice que la sentencia

$$\neg TT \vdash Q(\text{“}\neg TT \vdash Q(s)\text{”})$$

es inclasificable en la teoría TT. Podemos crear la teoría TTT a partir de la adición de la sentencia anterior como un axioma a TT. Sin embargo, el segundo teorema dice que la sentencia

$$\neg TTT \vdash Q(\text{“}\neg TTT \vdash Q(s)\text{”})$$

será inclasificable. Hablando informalmente, el segundo teorema es el mismo que el primer teorema, pero dejando el nombre de la teoría como una variable. A partir ahora no vamos a distinguir entre ellos.

6. Semántica e Intérpretes

Como hemos visto, algunas sentencias son teoremas, algunas antiteoremas y algunas ninguna de las dos. Puede tentar decir que las sentencias que no son ninguna de las dos están en una tercera clase, e inventar un tercer símbolo que vaya con $\vdash$ y $\dashv$. En algunos círculos, la lógica de tres valores es popular. Desafortunadamente, cualquier intento de formalizar la tercera clase correrá con el mismo problema: habrá un agujero, y la tentación de inventar una cuarta clase, y así sucesivamente. Yo prefiero decir que aquellas sentencias que que no son ni teoremas ni antiteoremas son “inclasificables”.

En esta sección presentaremos una metateoría más simple aunque menos expresiva que en las secciones previas, usando un solo símbolo. Para alguna teoría T y una cadena s introducimos la sentencia:

$$T \mathrel{\top\!\!\!\!\bot} s$$

El predicado $\top\!\!\!\!\bot$ representa interpretar la cadena s en la teoría T. Cuando esta claro de que teoría se habla, podemos omitir su nombre. Para alguna teoría, queremos que $\top\!\!\!\!\bot\, s$ sea un teorema si y sólo si s represente un teorema y que sea un antiteorema si y sólo si s represente un antiteorema. Está relacionado con $\vdash$ y $\dashv$ con las dos implicaciones:

$$\vdash s \Rightarrow \top\!\!\!\!\bot\, s \Rightarrow \neg \dashv s$$

De hecho, si hemos definido $\vdash$ y $\dashv$, estas implicaciones definen $\mathtt{I}$. Pero queremos $\mathtt{I}$ para reemplazar a $\vdash$ y $\dashv$ por lo que en lugar de ello debemos definirlo al mostrar cómo se aplica a toda forma de sentencia. Aquí está el principio de su definición.

$$
\begin{aligned}
&\mathtt{I}\ \text{“}true\text{”} = true\\
&\mathtt{I}\ \text{“}false\text{”} = false\\
&\forall s,\ \mathtt{I}\,(\text{“}\neg\text{”}s) = \neg\ \mathtt{I}\ s\\
&\forall s,\ \forall t,\ \mathtt{I}\,(s\text{“}\wedge\text{” t}) = \mathtt{I}\ s \wedge \mathtt{I}\ t\\
&\forall s,\ \forall t,\ \mathtt{I}\,(s\text{“}\vee\text{” t}) = \mathtt{I}\ s \vee \mathtt{I}\ t
\end{aligned}
$$

Y así sucesivamente. Tenga en cuenta que $\mathtt{I}$ actúa como el inverso de las comillas, "quitando las comillas" a su operando. Eso es lo que un intérprete hace: vuelve datos pasivos en un programa activo. Es un hecho familiar para los programadores que se puede escribir un intérprete de un lenguaje en el que el mismo lenguaje, y eso es precisamente lo que estamos haciendo aquí.

Para terminar de definir $\mathtt{I}$ tenemos que decidir los detalles de una teoría completa. No vamos a hacerlo aquí, pero vamos a dar un caso más de especial interés. $\mathtt{I}$ se define para cadenas que incluyen $\mathtt{I}$ como

$$\forall s,\ \mathtt{I}\,(\text{“}\mathtt{I}\ \underline{\text{“”}}\,s\,\underline{\text{“”}}\text{”}) = \mathtt{I}\ s$$

Así, el intérprete se convierte en parte de la lógica a interpretar.

7. Gödel Simplificado

Nuestra presentación del argumento de Gödel es paralela a la suya, pero usando cadenas en lugar de números. El argumento sería esencialmente el mismo si hubiéramos utilizado $\mathtt{I}$ en lugar de $\vdash$. El corazón del argumento es una transformación de un nivel de comillas a dos niveles, y de vuelta a uno, con un "$\neg$" que aparece en el proceso. Podemos simplificar el argumento, sin pérdida de contenido yendo de cero a uno y a la inversa. No tomamos Q como una función de cadenas a cadenas, sino simplemente como una cadena. Se define como:

$$Q = \text{“}\neg\ \mathtt{I}\ Q\text{”}$$

Con su adición no hacemos a la teoría incompleta: Q es igual a una cadena de caracteres de largo 3 completamente conocida. Ahora bien, supongamos que podemos construir (o definir) una función $\mathtt{I}$ "quitadora de comillas" (no importa cómo). Luego tenemos el teorema

$$
\begin{aligned}
&\mathtt{I}\ Q\\
&\{\text{reemplazando } Q \text{ por su igual}\}\\
&= \mathtt{I}\ \text{“}\neg\ \mathtt{I}\ Q\text{”}\\
&\{\text{debido a que } \mathtt{I} \text{ quita las comillas}\}\\
&= \neg\ \mathtt{I}\ Q
\end{aligned}
$$

y la inconsistencia es evidente. También lo son las propiedades necesarias para armar el argumento: una teoría que nos permita reemplazar algo por sus iguales, y debe incluir o permitir definir su propio intérprete. Para salvar a una teoría de la inconsistencia, se podría suspender la posibilidad de reemplazar algo por sus iguales, en determinadas circunstancias, pero esta es una opción desagradable. En su lugar, dejamos el intérprete incompleto. En particular, si

$$⌶ \text{“}\neg ⌶ Q\text{”} = \neg ⌶ Q$$

es un teorema tenemos inconsistencia, y si se trata de un antiteorema no es un intérprete, por lo que lo dejaremos como “inclasificable”.

Al igual que antes, debemos tener en cuenta que $⌶\ s$ es una sentencia para todas las cadenas s, incluso aquellas que no representan sentencias. De lo contrario, no podemos demostrar que $⌶\ Q$ es una sentencia. Es tentador interpretar Q como la representación de una sentencia que dice que ella misma (Q) no es interpretable. Pero en un sentido muy real, debemos abstenernos de interpretar de Q: en términos de programación, la aplicación del intérprete $⌶$ a la cadena Q sería provocar un bucle infinito de ejecución, y sabemos que no dan resultado. El argumento de Gödel es una versión del Problema de la Detención.

8. Conclusión

Después del trabajo pionero de Frege, Russell, y otros lógicos a principios de este siglo, David Hilbert esperaba que fuera posible formalizar toda la matemática. Pero en 1931, Kurt Gödel llegó a un resultado que se interpreta comúnmente diciendo que cualquier formalismo que incluya la aritmética nos permite expresar verdades de las matemáticas que no pueden ser probadas en el formalismo. Y con eso, la esperanza de Hilbert murió. El punto importante del resultado de Gödel no es la existencia de enunciados verdaderos, pero indemostrables; lo importante es que es fácil diseñar una teoría incompleta en la que algunas de las sentencias imposibles de demostrar pretendan representar verdades. El resultado de Gödel dice que no hay un formalismo que describe completamente todos los formalismos (incluido el mismo). Pero es igualmente cierto que cada formalismo es completamente descriptible por otro formalismo, y en ese sentido una esperanza de Hilbert más modesta es posible.

Creemos que nuestra presentación de los Teoremas de Incompletitud de Gödel ilustra muy bien la afirmación de E.W.Dijkstra de que la Ciencia de la Computación ahora puede pagar con intereses su deuda con las matemáticas. En concreto, la distinción entre el programa y los datos, el uso del tipo de dato cadena de caracteres, el uso de un intérprete, y la cuenta desde cero, redujeron la prueba a tres líneas.

Acuse de recibo. Doy las gracias a Bill McKeeman por sugerir que rescatara esta presentación del teorema de Gödel de un trabajo más largo y con inclinaciones filosóficas.

Referencias

[1] Gödel, K.: *Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I, Monatshefte für Mathematik und Physik*, v.38 p.173-198, Leipzig, 1931.

[2] Gödel, K.: *On Formally Undecidable Propositions Of Principia Mathematica And Related Systems*, traducido al inglés por B. Meltzer, con introducción de R.B. Braithwaite, Oliver & Boyd, Edimburgo, 1962.