THÉMATA. Revista de Filosofía
Nº 55, enero-junio (2017) pp.: 327-340.
ISSN: 0212-8365 e-ISSN: 2253-900X
doi: 10.12795/themata.2017.i55.15

TRES ARGUMENTOS ORDINALES CONTRA EL COMPUTACIONALISMO

THREE ARGUMENTS BASED ON ORDINALS AGAINST COMPUTATIONALISM

Laureano Luna Cabañero[1]
IES Doctor Francisco Marín, Jaén

Recibido: 14/ 4/ 16
Aceptado: 27/ 10/ 16

Resumen: Definimos Computacionalismo como la tesis de que la inteligencia humana es equivalente a un algoritmo. Esta tesis implica la tesis particular de que la inteligencia *matemática* humana es equivalente a un algoritmo. Asumiendo que todos los algoritmos son equivalentes a máquinas de Turing, argumentamos que, para cualquier algoritmo, la inteligencia matemática humana es capaz de generar y de definir ordinales que el algoritmo no es capaz de producir o definir. Terminamos con un argumento contra la posibilidad de que los algoritmos puedan definir en sentido estricto, es decir, incluyendo la dimensión semántica del concepto de definir. Consideramos además cómo quedan los argumentos si no suponemos que todos los algoritmos son máquinas de Turing.

Palabras clave: computacionalismo, algoritmo, máquina de Turing, inteligencia matemática humana, ordinales constructivos, semántica, sintaxis.

Abstract: We define Computationalism as the claim that human intelligence is equivalent to some algorithm. This claim entails the weaker claim that human *mathematical* intelligence is equivalent to some algorithm. On the assumption that all algorithms are equivalent to Turing machines, we argue, that for any given algorithm, the human mathematical intelligence is able to generate and define set theoretic ordinals the algorithm is not able to. We present in addition an argument against the possibility that algorithms are capable of defining in the strict sense, that is, including the semantical dimension involved in defining. Furthermore, we assess the value of the arguments when the assumption that algorithms are equivalent to Turing machines is dropped.

Key words: computationalism, algorithm, Turing machine, human mathematical intelligence, constructive ordinals, semantics, syntax.

1. (laureanoluna@yahoo.es) Licenciado en Lenguas Clásicas, doctor en Filosofía y profesor de Filosofía en el IES Doctor Francisco Marín, Siles (Jaén).

1. Introducción y definiciones

Cuando los psicólogos y filósofos de la mente se hicieron cargo de la capacidad de los algoritmos para imitar los procesos humanos de pensamiento algunos de ellos conjeturaron que el funcionamiento de la mente era de naturaleza computacional. Esa concepción de la naturaleza de la mente estaba ligada a las teorías sintácticas de Chomsky (Chomsky 1957) y al Funcionalismo de Putnam (Putnam 1960, donde se proponía la máquina de Turing como modelo de la mente humana) y se desarrolló en el seno del Cognitivismo (por ejemplo, Fodor 1984).

Fodor propone una teoría en la que las actividades cognitivas de la mente son de carácter computacional y se realizan en un sistema representacional de carácter algorítmico: el lenguaje del pensamiento, a veces llamado *mentalés*. El Computacionalismo proponía un proyecto más sutil de naturalización de la mente que el del Fisicalismo clásico, porque hacía equivaler los estados mentales no directamente a estados físicos del cerebro sino más bien al significado computacional, lógico o informacional de tales estados. En este contexto se asumía que el cerebro es funcionalmente un computador (tesis fundamentada en el trabajo seminal de McCullough y Warren 1943; para consideraciones críticas de esta tesis ver Penrose 1989, Searle 1990).

El Computacionalismo incluido en el Cognitivismo desembocó en la tesis de que es en principio posible que una máquina desarrolle un comportamiento cognitivo indistinguible de la actividad cognitiva humana, tesis sugerida por Turing (1950). Esa tesis es la que nuestros argumentos pretenden refutar. Nótese que la tesis de que la inteligencia humana es equivalente a un algoritmo es más fuerte que la tesis de que cada operación mental particular puede simularse algorítmicamente. Esta última tesis es casi trivial para algunos ámbitos; por ejemplo, parece evidente que cada tarea matemática particular que un ser humano realice puede escribirse y formalizarse y, una vez representada como una tarea sintáctica, puede ser desarrollada por un algoritmo. Esto, sin embargo, no implica que exista un algoritmo capaz de realizar él solo cuanto la inteligencia matemática humana es capaz de hacer. Por esta razón, la tesis del Computacionalismo en sentido fuerte es más interesante y es la que Lucas (1961), Penrose (1989), Bringsjord *et al.* (2006), etc. han tratado de refutar.

Dicho esto, sigamos con algunas definiciones.

En primer lugar, un algoritmo es un conjunto finito de instrucciones completamente especificadas, que pueden seguirse mecánicamente paso a paso para realizar una tarea sintáctica como escribir un símbolo, borrarlo, modificarlo o trasladarlo[2]. Una tarea realizada por un algoritmo suele recibir el nombre de 'computación'.

En segundo lugar, definimos Computacionalismo como la tesis de que la inteligencia humana es equivalente a un algoritmo. Esta tesis implica la tesis particular de que la inteligencia *matemática* humana es equivalente a un algoritmo. Esa equivalencia se entiende aquí como igualdad en la capacidad para desarrollar tareas matemáticas: definir, construir, demostrar...

Las capacidades matemáticas de las que hacemos depender los conceptos de *inteligencia matemática humana* y de *equivalencia* son tan amplias que podría pensarse que los conceptos no quedan suficientemente definidos para los propósitos de nuestra argumentación. Sin embargo, en cada una de las tres secciones que siguen ofrecemos un argumento contra el Computacionalismo basado en una capacidad concreta: la de generar ordinales (primer argumento) o la de definir ordinales (segundo y tercer argumentos). Creemos que esa concreción prestará a los conceptos de inteligencia matemática humana y de equivalencia la definición necesaria en cada caso para que el argumento resulte riguroso.

Introduzcamos ahora los ordinales conjuntistas. El primer ordinal conjuntista es el conjunto vacío, que se equipara con el 0. A partir de ahí, cada ordinal es el conjunto de todos los ordinales estrictamente menores que él:

$$\varnothing = 0;\ \{0\} = 1;\ \{0, 1\} = 2;\ \ldots$$

La pauta que nos da recursivamente lo ordinales finitos es pues:

$$0 = \varnothing;$$
$$n+1 = n \cup \{n\}.$$

En la teoría de conjuntos existen números ordinales que van más allá de los usuales ordinales finitos. Se llaman 'ordinales transfinitos'.

2. Considérese la definición de la noción informal de algoritmo en Rogers (1992, p. 1): "Roughly speaking, an algorithm is a clerical (i.e. deterministic, book-keeping) procedure which can be applied to any of a certain class of symbolic *inputs* and which will eventually yield, for each such input, a corresponding symbolic *output*".

Como cada ordinal es el conjunto de todos los anteriores, el primer ordinal transfinito es el conjunto de todos los ordinales finitos:

$\omega_0 = \{0, 1, 2, \dots\}$.

A partir de ahí

$\omega_0+1 = \omega_0 \cup \{\omega_0\}$,

y así indefinidamente. Estas pautas nos dan solo una parte de los ordinales transfinitos existentes en la teoría clásica de conjuntos (los ordinales enumerables) pero lo dejaremos aquí: nuestro argumento no requiere ir más allá (véase para más información, por ejemplo, Jech 2006, cap. 2).

Como todo ordinal es el conjunto de todos los ordinales estrictamente menores que él, ningún ordinal es elemento de sí mismo. La estructura de los ordinales es relativamente simple y eso sugiere que podría ser ventajoso basarse en las capacidades humanas de generar y de definir ordinales para argumentar contra el Computacionalismo de una forma más sencilla e intuitiva que usando los recursos habituales. Los argumentos contra el Computacionalismo basados en las capacidades matemáticas humanas han recurrido generalmente a la capacidad humana de demostrar teoremas para contraponerla a las limitaciones que los teoremas de Gödel o Turing imponen a los algoritmos que demuestran teoremas (Lucas 1961, Penrose 1989). Estos argumentos son complejos y controvertidos. Creemos que los que presentamos son como mínimo más sencillos e intuitivos.

2. Primer argumento

Este primer argumento intenta mostrar que la inteligencia matemática humana puede en principio generar más ordinales que cualquier algoritmo dado.

Sucede que hay algoritmos que pueden generar (nombres para) ordinales conjuntistas pautadamente desde el primer ordinal[3]. Cuando un algoritmo genere nombres pautadamente desde 0 hasta un nombre para un determinado ordinal α, diremos que el algoritmo genera o alcanza α. A los ordinales que son alcanzados o generados por algún algoritmo se llama ‘ordinales constructivos’. Hay ordinales constructivos infinitos, es decir,

3. En palabras de Church (1938, p. 224) se trata de ‘a step by step process of building up to the ordinal from below’. Otra manera de caracterizarlos es definir los ordinales constructivos como los ordinales recursivos (véase Rogers 1992 §11.8, corolario XX).

que contienen infinitos elementos. Cabe preguntarse cómo podría un algoritmo alcanzar un ordinal infinito. Como hemos visto a propósito de ω_0, la técnica para alcanzar ordinales infinitos es una especie de paso al límite: cuando un algoritmo tiene una subrutina capaz de generar un ordinal α y, para todo ordinal β generado por la subrutina, también el sucesor de β, decimos que la subrutina genera una *secuencia fundamental* S; entonces el algoritmo puede en principio generar el *supremo* de S, es decir, el menor ordinal mayor que todos los ordinales en S. El supremo de S es lo que se llama un *ordinal límite* y es el conjunto de todos los ordinales que están en S o son menores[4] que todos los elementos de S.

El conjunto de todos los ordinales constructivos es el ordinal ω_1^{CK}. Como ningún ordinal es elemento de sí mismo, ω_1^{CK} no es constructivo; es el primer ordinal no constructivo, el primero que ningún algoritmo alcanza[5]. Los ordinales constructivos y el ordinal ω_1^{CK} fueron definidos por Church y Kleene (Church, Kleene 1937; Church 1938; Kleene 1938; véase también Rogers 1992 §11.8). Church y Kleene, para definir el concepto de algoritmo (o *procedimiento efectivo*) utilizaron el cálculo lambda desarrollado por Church. Por tanto, y dado que lo lambda-computable es lo mismo que lo Turing-computable, por 'algoritmo' podemos entender en este contexto 'máquina de Turing'.

Es fácil ver que no existe ningún algoritmo que genere todos los ordinales constructivos. Si lo hubiera, sería posible construir a partir de él un algoritmo que generase también ω_1^{CK}, y entonces tendríamos la contradicción de que el primer ordinal no constructivo sería constructivo. Para todo ordinal constructivo hay al menos un algoritmo que lo alcanza pero no hay un algoritmo que alcance todos los ordinales constructivos. Por tanto, para cada algoritmo que genera ordinales constructivos hay otro que genera más ordinales constructivos. Y esto es así por necesidad lógica.

Debido a que todo algoritmo tiene una definición precisa y finita, entendemos que la inteligencia humana puede *en principio* usar cualquier algoritmo y desarrollar la tarea indicada en sus instrucciones. Esto significa que no hay imposibilidad lógica o matemática (imposibilidad *por principio*) para ello aunque puedan existir otras clases de imposibilidad para que una

4. El 'o menores' es necesario porque un ordinal contiene todos los ordinales menores que él y S puede dejar fuera ordinales menores que algunos de los que están en S; considérese la secuencia 1,2, 3,... Nuestra formulación permite evitar el concepto más técnico de *clausura transitiva*; por eso, usamos una formulación similar más abajo en (2).

5. Desde un punto de vista constructivista el ordinal ω_1^{CK} podría no existir: podría ser el universo ordinal en lugar de un ordinal.

mente humana particular use determinado algoritmo y desempeñe la correspondiente tarea: imposibilidad técnica, física, cultural, biológica, etc.

Hay que subrayar que cuando decimos que un algoritmo puede desempeñar una tarea nos referimos con frecuencia también a su capacidad *en principio*. Por ejemplo, admitimos que existe un algoritmo capaz de enumerar los números naturales (es decir, capaz, para cualquier natural n, de generar n en un número finito de pasos), a pesar de que se trata de una tarea infinita para lo que no existe papel o tinta o energía eléctrica suficiente. Y es que no concebimos los algoritmos como máquinas físicas que usan determinados materiales, consumen cierta cantidad de energía y necesitan cierta cantidad de tiempo para cada paso, sino como objetos matemáticos abstractos, como conjuntos de instrucciones. De igual manera, debemos entender que la capacidad de la inteligencia matemática humana está limitada solo por los límites en la información lógico-matemática que esta inteligencia pueda contener.

Así, por ejemplo, admitimos que la inteligencia humana puede sumar cualquier número finito de números naturales (porque hay un algoritmo que lo hace) aunque en la práctica nos resulte imposible sumar un trillón de números de un trillón de dígitos cada uno, al menos si se eligen adecuadamente esos números. De hecho, el resultado de esa suma es una consecuencia lógica de la aritmética de Peano de primer orden, cuyos axiomas hemos construido nosotros y para la que tenemos una lógica completa (capaz de deducir las consecuencias lógicas de cualquier conjunto de sentencias).

Como para cada algoritmo que genera ordinales hay otro que genera más ordinales y la inteligencia matemática humana puede en principio usar cualquier algoritmo, la inteligencia matemática humana puede en principio hacer lo que ningún algoritmo por principio puede: generar más ordinales que cualquier algoritmo dado. En consecuencia, la inteligencia humana no equivale a algoritmo alguno y el Computacionalismo es falso.

3. Segundo argumento

Nuestro segundo argumento se basará no en la capacidad para generar ordinales (la capacidad para generar nombres para ordinales pautadamente desde 0) sino en la capacidad para *definir* ordinales e intentará mostrar que, para cada algoritmo dado, la inteligencia matemática humana es capaz de definir un ordinal que el algoritmo no es capaz de definir. La inteligencia matemática humana es capaz de definir ordinales. Por ejemplo, esta fórmula

(1) $0 = \{x \neq x\ ^{1}\ x\}$

define el primer ordinal.

Si, como el Computacionalismo pretende, la inteligencia matemática humana equivale a un algoritmo, entonces hay algoritmos capaces de definir ordinales, dado que (1) atestigua que la inteligencia matemática humana puede definir ordinales. Supongamos entonces que hay algoritmos capaces de definir ordinales. Asumiremos además la llamada *tesis de Church-Turing,* que afirma que todo lo computable es Turing computable, de modo que podremos asumir sin pérdida de generalidad que todo algoritmo es una máquina de Turing.

Sea ALG un algoritmo cualquiera capaz o incapaz de definir ordinales. Ese algoritmo es, como todos, finitamente definible. De hecho (asumiendo, como hacemos, la tesis de Church-Turing) el conjunto de todos los algoritmos es algorítmicamente enumerable, lo que significa que hay un algoritmo ALG_E que los enumera a todos (véase por ejemplo Hamilton 1981, pp. 189-190). La inteligencia humana puede por tanto en principio usar una enumeración recursiva de todos los algoritmos e ir nombrándolos uno a uno; tarde o temprano llegará a nombrar a ALG. Como consecuencia de esto, la inteligencia matemática humana puede en principio definir para cada ALG un ordinal mayor que cualquier ordinal definido por ALG de la siguiente manera.

Definimos primero el conjunto O_{ALG} de todos los ordinales iguales a o menores que algún ordinal definido por ALG:

(2) $O_{ALG} = \{x \mid \exists y\ (y \in Ord\ \&\ D(ALG, y)\ \&\ x \leq y)\}$,

donde ‘Ord’ denota la clase de todos los ordinales y ‘D(ALG, y)’ dice que ALG define y. O_{ALG} es un ordinal mayor que todo ordinal definido por ALG ya que si O_{ALG} fuese definido por ALG se contendría a sí mismo y, como hemos visto, ningún ordinal se contiene a sí mismo.

Nótese que si ALG no define ordinales, entonces, por (2), $O_{ALG} = \varnothing = 0$, y mediante (2) conseguimos igualmente definir un ordinal que ALG no define. Por tanto, el argumento no requiere que sea decidible si un algoritmo dado es o no capaz de definir ordinales[6].

Parece, por consiguiente, que si hay algoritmos capaces de definir ordinales, entonces, para cada algoritmo dado, la inteligencia humana puede en principio definir un ordinal que ese algoritmo no define. Como eso, por principio, no puede hacerlo ningún algoritmo (porque, si pudiera, al aplicarse a sí mismo definiría un ordinal que él mismo no define), la inteligencia humana no es equivalente a un algoritmo. Y, si no hay algoritmos capaces de definir ordinales, parece igualmente claro que ningún algoritmo es equivalente a la inteligencia humana, que sí es capaz de hacerlo.

Cabría, sin embargo, objetar que, dado un algoritmo ALG, puede no estar bien definido[7] si ALG define o no algún ordinal, de modo que no esté bien definido si $O_{ALG} = 0$ (que sería el caso si ALG no define ordinales o solo define 0) o $O_{ALG} \neq 0$ (en caso contrario). Si eso fuera así, la definición de O_{ALG} sería incapaz de definir un ordinal. Pero esto sugiere inmediatamente la siguiente consideración: el que, dado un algoritmo ALG, no esté bien definido si ALG define o no ordinales solo puede ser el caso si definir ordinales no es una función inherente a las instrucciones de ALG sino que depende de que cómo se interpreten esas instrucciones o sus resultados.

En efecto, podría entenderse que un algoritmo que generase esta secuencia de símbolos:

$0 = \{x \mid x \neq x\}; 1 = 0 \cup \{0\}; 2 = 1 \cup \{1\}; \ldots$

estaría definiendo ordinales para algunos observadores (aquellos que interpretan esos símbolos de la manera usual) y no estaría definiendo ordinales para otros: la interpretación que dota a los objetos sintácticos de una dimensión semántica podría ser ingrediente necesario para que tenga lugar lo que, en sentido estricto, entendemos por definir ordinales. Ahora bien, si esto fuera así, sería muy dudoso que los algoritmos pudieran en sí mismos definir ordinales como lo hace la inteligencia matemática humana. Los algoritmos son criaturas sintácticas y no está claro que la semántica pueda reducirse a sintaxis a pesar de los intentos de resolver los problemas implicados en tal reducción (véase por ejemplo Taddeo, Floridi 2005 para el problema del *symbol grounding*).

Esto nos conducirá a nuestro tercer argumento.

Pero aclaremos antes este punto: ¿cómo quedaría este segundo argumento si dejamos de admitir que todos los algoritmos son máquinas de

6. Que este problema es indecidible (bajo la tesis de Church-Turing) se sigue del teorema de Rice (debo esta observación a un revisor anónimo).

7. 'No definido' no en el sentido de que sea algorítmicamente indecidible sino en el sentido más fundamental de que el principio de Bivalencia ('p v ¬p') no se aplique porque la cuestión no esté matemáticamente bien definida.

Turing? Este cambio en las premisas podría desde luego afectar al argumento, porque de las máquinas de Turing hay una enumeración algorítmica, de modo que la inteligencia matemática humana puede en principio dar con cualquiera de ellas y definir el correspondiente O_{ALG} pero nada nos asegura que, si hay algoritmos que no son (equivalentes a) máquinas de Turing, haya también una enumeración algorítmica de todos ellos. Si existiesen algoritmos que la inteligencia humana no puede llegar a conocer y nombrar, este argumento y el anterior fallarían: uno de esos algoritmos podría ser equivalente a la inteligencia humana a pesar de nuestro argumento, porque a la inteligencia matemática humana le resultaría imposible por principio definir el correspondiente O_{ALG}. Ahora bien, dado que la definición de algoritmo establece que se tratará siempre de un conjunto *finito* de instrucciones resulta cuando menos inverosímil que existan algoritmos incognoscibles por principio para la inteligencia humana.

4. Tercer argumento

Trataremos de hacer ver que, en sentido estricto, ningún algoritmo es capaz de definir ordinales tal como lo hace la inteligencia matemática humana.

Decíamos que los algoritmos son de naturaleza puramente sintáctica y que la acción de definir tiene una dimensión semántica, de modo que si la semántica involucrada en la tarea humana de definir ordinales no pudiera reducirse a sintaxis, el Computacionalismo sería falso[8].

Este tercer argumento es una consecuencia del argumento anterior. Informalmente podemos exponerlo así: hemos visto en el argumento anterior que existe una pauta uniforme que permite, dado un algoritmo ALG cualquiera, definir un ordinal O_{ALG} que ALG no define; si los algoritmos pudieran definir ordinales, habría un algoritmo capaz de utilizar esa pauta para definir O_{ALG} para cada ALG, incluido él mismo, y esto llevaría a una contradicción[9].

Formalmente, el argumento es una reducción al absurdo. Seguiremos por ahora bajo la suposición de que todos los algoritmos son máquinas de Turing. Supongamos pues que de alguna manera toda la semántica que la inteligencia humana utiliza en la tarea de definir ordinales puede reducirse a sintaxis y que, por tanto, hay algoritmos que definen ordinales. Sabemos que hay un algoritmo ALG_E que enumera a todos los algoritmos.

8. Para argumentos de carácter semántico contra el Computacionalismo, véase Searle 1994, Luna 2014.

9. Hay razones independientes para pensar que la capacidad semántica humana no es de naturaleza algorítmica; véase Luna 2013.

Entonces ALG_E podría utilizarse para construir un algoritmo ALG_E* capaz de definir O_{ALG} para cada algoritmo ALG enumerado por ALG_E: al fin y al cabo, la definición de O_{ALG}, dado ALG, sigue una pauta uniforme. Por tanto, para todo algoritmo ALG, ALG_E* definiría un ordinal O_{ALG} que ALG no define. Como ALG_E* es uno de los ALG, ALG_E* definiría un ordinal que ALG_E* no define, lo que es una contradicción.

Naturalmente, si los algoritmos no pueden en sentido estricto definir ordinales como hace la inteligencia matemática humana, ningún algoritmo representa adecuadamente la inteligencia matemática humana y el Computacionalismo es falso.

Aunque este argumento, tal como se ha expuesto, emplea la premisa de que todos los algoritmos son equivalentes a máquinas de Turing, es fácil ver que para el argumento basta asumir que el conjunto de todos los algoritmos es algorítmicamente enumerable (sean sus elementos máquinas de Turing o no). Si hubiese algoritmos que no son equivalentes a máquinas de Turing y si, como consecuencia, el conjunto de todos los algoritmos *no* fuese algorítmicamente enumerable, nuestro argumento no valdría.

Sin embargo, si nuestro tercer argumento realmente demuestra en concreto de las máquinas de Turing que no pueden tener la capacidad semántica implícita en la función de definir, cabe preguntarse cómo podrían tener esa capacidad otros algoritmos. Es natural pensar que un algoritmo, incluso si no fuera equivalente a una máquina de Turing, no podría ser esencialmente diferente de las máquinas de Turing a ese efecto: seguiría siendo, como ellas, un conjunto finito de instrucciones bien especificadas que permiten realizar de manera mecánica una tarea de índole sintáctica.

5. Conclusiones

Los argumentos que hemos ofrecido sugieren dos consideraciones finales.

Por una parte, todo algoritmo es, por definición, un objeto bien definido para la inteligencia matemática humana; un objeto, por tanto, en el que esta inteligencia puede en principio apoyarse para generar o definir ordinales que el algoritmo no genera o no define; obviamente ningún algoritmo puede hacer tal cosa con respecto a sí mismo. En este preciso sentido, ningún algoritmo es un objeto bien definido para sí mismo aunque todos los son para la inteligencia matemática humana. Este hecho podría establecer una diferencia esencial entre unos y otra.

Por otra parte, el tercer argumento sugiere que la tarea matemática que desarrolla la inteligencia humana al definir objetos matemáticos tiene una dimensión semántica que no puede reducirse a ninguna de las tareas sintácticas que un algoritmo es capaz de realizar.

Referencias bibliográficas

Bringsjord, S.; Kellet, O.; Shilliday, A.; Taylor, J.; Van Heuveln, B.; Yang, Y.; Baumes, J.; Ross, K. 2006. "A new Göddelian argument for hypercomputing minds based on the Busy Beaver problem" en *Applied Mathematics and Computation* 176: 516-530.

Church, A. 1938. "The constructive second number class" en *The Bulletin of the American Mathematical Society* 44(4): 224-232.

Church, A.; Kleene, S. C. 1937. "Formal definitions in the theory of ordinal numbers" en *Fundamenta mathematicae* 28: 11-21.

Fodor, J. A. 1984. *El lenguaje del pensamiento*. Madrid: Alianza Edtorial.

Hamilton A. G. 1981. *Lógica para matemáticos*. Madrid: Paraninfo.

Jech, T. 2006. *Set theory*. Springer: Berlín.

Kleene, S. C. 1938. "On notation for ordinal numbers" en *The Journal of Symbolic Logic* 3(4): 150-155.

Lucas, J. R. 1961. "Minds, machines, and Gödel" en *Philosophy,* 36(137): 112-27.

Luna, L. 2014. "Minds vs. Machines: on Saka's Basic Blindspot Theorem" en *Journal of Experimental and Theoretical Artificial Intelligence* 27(4): 483-486.

Luna, L. 2013. "Indefinite extensibility in natural language" en *The Monist* 96(2): 295-308.

McCullough, W. S., Pitts, W. 1943. "A logical calculus of the ideas immanent in nervous activity" en *Bulletin of Mathematical Biophysics* 5: 115-133.

Penrose, R. 1989. *The emperor's new mind: concerning computers, minds, and the laws of physics*. Oxford: Oxford University Press.

Putnam, H. 1960. "Mentes y máquinas" en Anderson A. R: *Controversias sobre Mentes y Máquinas*. Barcelona: Tusquets, 1984, pp. 113-149.

Rogers, H. 1992. *Theory of recursive functions and effective computability*. Cambridge (MA): The MIT Press.

Searle, J. 1994. *Mentes, cerebros y ciencia*. Madrid: Cátedra.

Searle, J. 1990. "Is the brain a digital computer?" en *Proceedings and Addresses of the American Philosophical Association* 64(3): 21-37.

Taddeo, M; Floridi, L. 2005. "Solving the symbol grounding problem: a critical review of fifteen years of research" en *Journal of Experimental and Theoretical Artificial Intelligence* 17 (4): 419-445.

Turing, A. 1950. "Computing machinery and intelligence" en *Mind* 59(236): 433-460.

RESEÑAS BIBLIOGRÁFICAS