EL CONCEPTO COMPUTACIONAL DE LA MENTE[1]

Halley D. Sánchez

Que el funcionamiento de la mente humana es análogo al funcionamiento de una computadora digital es una tesis ampliamente difundida hoy día. La misma, la cual supone que el pensar es una especie de computación, constituye la base de las investigaciones de personas en distintas especializaciones que trabajan en temas agrupados de forma general bajo el nombre de ciencias cognoscitivas. Ingenieros, científicos de la computación, matemáticos, biólogos, lingüistas, psicólogos y filósofos, entre otros, colaboran en lo que representa uno de los principales quehaceres intelectuales de nuestro tiempo.

Para poder discurrir sobre el concepto computacional de la mente es menester tener una noción, por lo menos general, de la teoría moderna de computación. Esta teoría surge en el siglo XX, particularmente en relación al intento de usar técnicas de la (para aquel entonces) recién descubierta/inventada lógica matemática (o simbólica) para aclarar algunas cuestiones de los fundamentos de las matemáticas. La noción de algoritmo es central a la teoría moderna de computación. Un algoritmo es un procedimiento general mecánico que resulta en una conclusión o solución, de haberla, en un número de pasos determinados. Cada paso de un algoritmo está completamente prescrito y no puede haber duda de cuál ha de ser el siguiente paso. Un algoritmo es general en el sentido de servir para resolver todos los problemas de una clase específica de problemas, aunque, por supuesto, para diferentes clases de problemas

[1] Una versión abreviada y preliminar de este trabajo fue presentado el 3 de mayo de 1993 a estudiantes de Filosofía de la Psicología en el Recinto Universitario de Mayagüez de la Universidad de Puerto Rico. Deseo agradecer a la National Endowment for the Humanities por haberme concedido una beca “Summer Seminar for College Teachers” (Universidad de Maryland, verano 1992, dirigido por Michael D. Resnik) durante la cual llevé a cabo investigaciones en la filosofía de las matemáticas que sirvieron como base para los estudios que aquí se presentan.

habrá que haber diferentes algoritmos. Un modo de entender el concepto de algoritmo en el sentido preciso matemático es a través de la llamada máquina de Turing, concepto matemático idealizado introducido por el matemático y lógico Alan Turing (1912-1954) en el año 1935-36. Pero antes de considerar el concepto de Turing, conviene repasar el trasfondo histórico del concepto.

El término algoritmo se deriva del nombre del matemático árabe Al-JWARIZMI, quien vivió alrededor del año 800, aunque la noción de un algoritmo es aún más antigua, quizá tan antigua como las matemáticas mismas. Inclusive las instrucciones para construir ciertas figuras, como las que se encuentran en *Los Elementos* de Euclides, son especies de algoritmos. Por ejemplo, la Proposición 1 del Libro I de Euclides explica cómo, utilizando solamente una regla para tirar líneas rectas y un compás, dada cualquiera línea recta finita se puede construir un triángulo equilátero cuyos lados son iguales a la línea recta inicial; la Proposición 3 explica cómo, dada dos líneas rectas desiguales, se puede restar de la línea recta más extensa una cantidad igual a la línea menos extensa. Cada una de estas instrucciones es un algoritmo. Pero fue con el álgebra y los matemáticos árabes que la noción de algoritmo comenzó a hacerse explícita con el desarrollo de procedimientos generales para resolver problemas algebraicos.

Uno de los primeros pensadores en hablar de (o soñar con) la mecanización del pensamiento fue el monje franciscano español Raimundo Lulio (1232-1316). Inspirado, según indica Hans Hermes (1984, p. 62), por los métodos matemáticos introducidos por los árabes, Lulio afirmó en su obra *Ars Magna* que el razonamiento se realiza no a base de silogismos (lógica aristotélica) sino a base de la descomposición y recombinación de representaciones, es decir, por combinatoria. En el siglo XVII, el filósofo y matemático Gottfried Wilhelm Leibniz (1646-1716), co-inventor del cálculo infinitesimal, propuso el desarrollo de un método mecánico universal, un *matesis universalis*, con lo cual todo enunciado podría ser expresado formalmente y comprobado en cuanto a si era verdadero o falso. Leibniz no produjo tal método universal, pero, además de legarnos la idea (el sueño) de tal método, logró construir una máquina calculadora que podía sumar, restar, multiplicar y dividir. En cuanto a máquinas calculadoras, anterior a Leibniz el filósofo y matemático francés Blaise Pascal (1623-1662) había construido una máquina calculadora (hacia 1642-1644), pero ésta sólo podía sumar y restar; y aún antes de Pascal, Wilhelm Schickard (1592-1635), un profesor de astronomía, matemáticas y hebreo en la Universidad de Tubingen, Alemania, había diseñado y supuestamente construido una máquina calculadora que sumaba y restaba de forma automática, y multiplicaba y dividía de forma semi-automática (Goldstine, 1972/1993, p. 6). Sin embargo, sin querer restarle importancia al ingenio para la construcción de máquinas calculadoras, hay que notar que la noción de un método para expresar formalmente todo posible enunciado junto con un algoritmo o procedimiento decisional para éstos es una idea mucho más profunda y abarcadora que la construcción de una máquina que ejecuta algunos algoritmos simples y específicos como son las cuatro operaciones básicas de la aritmética. Pero, para poder pasar del sueño a la proximidad de la realidad de un método universal, se requirió un avance significativo en la lógica: fue necesario que se descubriera lo que se conoce como lógica matemática.

Hacia mediados del siglo XIX, George Boole (1815-1864) desarrolló la aplicación de algunos de los métodos y leyes del álgebra a ciertas conexiones lógicas, lo cual representa un paso definitivo en el camino hacia la matematización y, por lo tanto, mecanización del razonamiento. Sin embargo, a pesar de que la algebraización de Boole representó un paso significativo en el avance de la lógica, de que su uso de algunos conectores ayudó a sentar la base para lo que hoy día se conoce como lógica de enunciados, de que su manejo de la noción de clases fue importante para la futura teoría de conjunto y de que la llamada álgebra booleana es de gran utilidad hoy día en la ingeniería eléctrica en cuanto al análisis de circuitos y el diseño de “microchips”, la lógica de Boole en sí no presenta una estructura clara de lo que constituye una prueba formal y el poder expresivo de esta lógica es muy limitado. Es decir, en la lógica booleana solamente es posible expresar en forma simbólica una parte limitada de lo que se consideran argumentos válidos. Por ejemplo, no es posible expresar en lógica booleana los argumentos geométricos de Euclides ni otros argumentos matemáticos más sofisticados (Glymour, 1992, pp. 109-112). Era necesario desarrollar una teoría de prueba utilizando una especie de estructura axiomática como se encuentra en la geometría y era necesario inventar/introducir procedimientos adicionales que permitan expresar en forma simbólica aseveraciones, argumentos y teorías complicadas. Estos avances pueden atribuirse a varias personas, pero sobre todo a Gottlob Frege (1848-1925), Alfred North Whitehead y Bertrand Russell. Los últimos dos escribieron lo que muchos consideran la obra monumental de la lógica matemática, *Principia Matematica* (3 vols., 1910-1913).

Parte importante de los logros de Frege, Whitehead y Russell fue mostrar que una parte significativa de las matemáticas puede formularse de manera explícita con la ayuda de la lógica simbólica. Esto permitió a algunos pensadores, como por ejemplo al matemático David Hilbert (1862-1943), hacer uso de los medios que provee la lógica en su intento de resolver algunas dudas que habían surgido hacia finales del siglo XIX y comienzos del siglo XX en cuanto a los fundamentos de las matemáticas. Esto es importante para nuestro tema, ya que, como se había indicado brevemente al comienzo del trabajo, algunos estudiosos de los orígenes de la ciencia cognoscitiva (Aspray, 1980) sostienen que la teoría moderna de computación surge principalmente en relación al esfuerzo de aclarar los fundamentos de las matemáticas. Explicar en detalles las dudas que surgieron en los fundamentos de las matemáticas y explicar el programa de Hilbert para intentar resolver las mismas está más allá del alcance de este trabajo. En términos generales, el programa de Hilbert consistía de: formular una teoría en cuestión usando el lenguaje de la lógica simbólica; aritmetizar el lenguaje para que cuestiones de las propiedades de la teoría estuvieran representados por relaciones aritméticas; establecer (probar la existencia de) un procedimiento constructivo que determinara para cada enunciado de la teoría si era o no era un teorema de la teoría; probar que la teoría en cuestión es consistente porque no contiene como teorema ningún enunciado y su negación. Irrespectivamente si tuvo o no éxito el programa de Hilbert (la opinión prevaleciente es que el mismo perdió su efectividad por causa de los llamados teoremas de incompletud de Gödel), resulta que el programa de aclarar los fundamentos de las matemáticas, por lo menos en la versión de Hilbert, requirió determinar precisamente lo que es y no es computable, y este esfuerzo formó la base de la teoría moderna de computación, que a su vez formó la base de los conocimientos para la construcción de las computadoras electrónicas, cuya construcción dio y da impulso a la teoría computacional de la mente.

Uno de los que contribuyó a la tarea de precisar lo que significa ser computable fue el matemático y lógico Alan Turing, quien se mencionó al comienzo del trabajo en relación a la llamada máquina de Turing, a saber, un aparato imaginario e idealizado que representa una de las formas de entender el concepto de algoritmo en el sentido preciso matemático. La descripción precisa de la máquina de Turing también está fuera del alcance de este trabajo, pero la misma puede describirse de modo informal como sigue. La máquina cuenta con una cinta potencialmente ilimitada para la entrada, el almacenaje y el reporte de información. Esta cinta está dividida en cuadrados, cada uno de los cuales puede estar vacío o puede tener un símbolo impreso. La máquina en sí tiene la capacidad de leer lo que está impreso en uno y sólo uno de los cuadrados a la vez. Además de leer lo que está impreso, la máquina tiene la capacidad de borrar o no borrar lo que está impreso, de escribir otro símbolo en el cuadrado, de moverse un cuadrado a la vez, ya sea a la izquierda o a la derecha, y de detenerse o parar. La máquina también cuenta con un número finito de estados internos que en combinación con la descripción de los símbolos que puede leer/escribir y con las movidas que puede hacer (derecha, izquierda, parar) completamente describen la máquina. Consideremos unos ejemplos sencillos.[2]

Máquina #1 - Digamos que el único símbolo que puede estar en la cinta es un 1, tal que al encontrar un cuadrado la máquina o bien se encontrará con un 1 ó bien con el cuadrado en blanco (para poder representar numéricamente en la tabla el cuadrado en blanco usemos un 0). Digamos que la máquina cuenta con dos estados internos, que también representaremos con 0 ó 1. Usemos las siguientes letras para representar el movimiento de la máquina: I = moverse un cuadrado a la izquierda; D = moverse un cuadrado a la derecha; P = parar. La siguiente tabla describe una máquina simple:

Estado inicial	**Símbolo**	**Escribir**	**Mover**	**Estado final**
0	0	0	D	0
0	1	1	D	1
1	0	1	D/P	—
1	1	1	D	1

Esta máquina se moverá por la cinta hasta encontrarse con un grupo de unos, y entonces le añadirá un uno (1) adicional al grupo antes de parar. Así que si se comienza con un cinta como la siguiente

		1	1	1	1											

la máquina producirá una cinta como la que sigue.

		1	1	1	1	1										

Consideremos una máquina un poco más complicada (máquina #2) usando las mismas definiciones que usamos para la máquina

2 Los siguientes ejemplos están inspirados parcialmente por algunos de los ejemplos que provee Roger Penrose en su libro *The Emperor's New Mind* (Penrose, 1989, pp. 38-46).

anterior con la excepción de que esta máquina requiere seis estados internos (denominados 0-5).

Estado inicial	**Símbolo**	**Escribir**	**Mover**	**Estado final**
0	0	0	D	0
0	1	0	D	1
1	0	1	I	2
1	1	1	D	1
2	0	0	D	3
2	1	0	D	4
3	0	1	D/P	—
3	1	1	D	3
4	0	1	I	5
4	1	1	D	4
5	0	1	I	2
5	1	1	I	5

La función de esta máquina es la de duplicar el número de unos que se encuentran agrupados en la cinta, irrespectivamente de la cantidad de unos con los cuales se encuentra. En otras palabras, la máquina representa la función de multiplicar por dos. Así que si la máquina se encontrara con la siguiente cinta

		1	1	1												

pasaría por los siguientes pasos intermedios hasta llegar al resultado final (para que se puedan seguir los pasos de la máquina se ha indicado con un asterisco (*) el cuadrado donde se encontraría la máquina en cada paso):

Paso #

1.			*1	1	1												
2.				*1	1												
3.				1	*1												
4.				1	1	*											
5.				1	*1	1											

6.				1		*1											
7.				1		1	*										
8.				1		*1	1										
9.				1	*	1	1										
10.				*1	1	1	1										
11.					*1	1	1										
12.					1	*1	1										
13.					1	1	*1										
14.					1	1	1	*									
15.					1	1	*1	1									
16.					1	*1	1	1									
17.					*1	1	1	1									
18.				*	1	1	1	1									
19.			*	1	1	1	1	1									
20.				*1	1	1	1	1									
21.				1	*1	1	1	1									
22.				1	1	*1	1	1									
23.				1	1	1	*1	1									
24.				1	1	1	1	*1									
25.				1	1	1	1	1	*								
26.				1	1	1	1	1	1	*							

De modo correspondiente se pueden confeccionar máquinas semejantes que computan otras funciones que permiten solucionarse de forma recursiva o algorítmica, aunque para esto conviene usar sistemas de símbolos más sofisticados y económicos, como, por ejemplo, el acostumbrado uso del sistema binario para representar números y entrar data a la máquina (por razones heurísticas no se utilizó el sistema binario para los dos ejemplos anteriores). Además, si se utiliza un sistema para numerar cada máquina particular y para entrar el número de la máquina deseada como parte de la data, puede idearse una máquina capaz de emular todas las posibles máquinas de Turing, a saber, una máquina universal de Turing.

La importancia del concepto de la máquina de Turing para la teoría de computación está resumido en lo que se conoce como la **tesis de Church-Turing**[3] según la cual el concepto de máquina de Turing (o su equivalente)[4] capta en el sentido matemático el significado de un proceso algorítmico o recursivo. En otras palabras, si ser computable equivale a poder ser computado por una máquina de Turing, entonces la máquina de Turing capta el significado de lo "computable" como tal. Además, a pesar de que la máquina de la cual habló Turing es un concepto matemático abstracto y que Turing desarrolló su concepto años antes del desarrollo técnico de las computadoras digitales electrónicas, para nosotros, acostumbrados a la presencia imperante en la vida cotidiana de las computadoras digitales, la relación entre la máquina de Turing y las computadoras actuales es obvia: en palabras de Nicanor Ursúa, "en cierto modo, el computador de los muchos que existen en el mercado es la realización física de una máquina universal de Turing" (1990, p. 60).

El hecho de tener una noción explícita del significado de computación, y el desarrollo de máquinas digitales electrónicas programables capaces de realizar algoritmos, fortalecieron la suposición de que el pensar es una especie de computación o proceso algorítmico y, por consiguiente, que el funcionamiento de la mente humana es análogo al funcionamiento de una computadora digital. Como se dijo al comienzo de este trabajo, este postulado es la base y el hilo

---

3 Alonzo Church, lógico matemático de origen norteamericano.

4 Hacia el tiempo que Turing desarrolla su concepto de máquina de Turing, otros lógicos matemáticos, entre ellos Alonzo Church y el polaco-norteamericano Emil Post, desarrollan esquemas para precisar lo que es computable. Sin embargo, se supone que todos estos esquemas son idénticos, afirmación que los estudiosos tienden a incluir como parte de la tesis Church-Turing.

conductor de las distintas investigaciones que se agrupan bajo el nombre de ciencias cognoscitivas. Por supuesto, nadie supone que el cerebro humano sea una computadora digital electrónica, pues el cerebro está compuesto de neuronas y no de piezas electrónicas como capacitadores, transistores y circuitos integrados. Pero dado que la tesis Church-Turing alega que toda descripción de los procesos algorítmicos son fundamentalmente idénticos y que los mismos pueden resumirse en términos de máquinas de Turing, el hecho que el cerebro humano es un órgano bioquímico y no una máquina electrónica con microcircuitos de silicio no le resta importancia a su posible funcionamiento algorítmico. Así que un aspecto significativo de las ciencias cognoscitivas está dedicado a encontrar algoritmos que simulen las funciones que lleva a cabo la mente humana.

Ahora bien, si no hay demasiada controversia en cuanto a la utilización heurística de la simulación computarizada para ayudar a entender el funcionamiento (o parte del funcionamiento) de la mente humana, sí hay controversia en cuanto a si una máquina es capaz de "pensar" y de "entender", de funcionar como una "mente" en el sentido "humano". Esta controversia divide a los defensores de la inteligencia artificial (IA) en el llamado sentido "fuerte" de los defensores de la IA en el sentido "débil" (y/o opositores de la IA). Actualmente también se tiende a distinguir dos interpretaciones de la IA fuerte, la clásica y la no-clásica, pero antes de introducir esta última distinción hay que tratar de la división más tradicional.

Uno de los defensores tradicionales de la IA fuerte fue el mismo Turing. En un artículo publicado en 1950, Turing, en vez de contestar directamente la pregunta sobre si una máquina puede pensar, substituyó la pregunta por lo que él llamó el "juego de imitación" y que actualmente se conoce como la "prueba de Turing" ("Turing-test"). La "prueba de Turing" consiste en establecer un intercambio o especie de "conversación" entre dos partes, una de las cuales ha de ser un ser humano que actúa como interrogador e intenta acertar si la otra parte del intercambio es un ser humano o una máquina. Por supuesto, el intercambio debería llevarse a cabo por escrito, por ejemplo, mediante terminales de computadoras o medios semejantes, y las dos partes del intercambio deberían estar en cuartos separados. Cuando los interrogadores no son capaces de acertar consistentemente cuándo están conversando con una máquina, se "podrá hablar del pensamiento de las máquinas sin esperar que se le contradiga" (Turing, 1985, p. 32). De un solo golpe Turing, con su juego de imitación, transcendió una variedad de objeciones que puedan levantarse en contra de la tesis de que las máquinas piensan (en su artículo varias objeciones fueron discutidas una a una).

Inclusive la difícil cuestión de si una máquina tiene conciencia fue esquivada por Turing con la observación de que la cuestión involucraría el espectro del solipsismo.

Una objeción en contra del programa de IA fuerte que no fue anticipada por Turing es la que se conoce como el "cuarto chino", un experimento pensado (*Gedankenexperiment*) presentado por el filósofo norteamericano John Searle que emplea el mismo juego de imitación de Turing para respaldar una conclusión contraria a la de Turing. El experimento pensado trata de un hombre que no conoce chino encerrado en un cuarto. Además de los medios de sobrevivir, al hombre se le ha provisto de una especie de manual que le permite intercambiar una serie de caracteres chinos que se le suministran por otros caracteres chinos que él ha de devolver. Así que cuando un interrogador le entrega (por una apertura en la pared) un papel con ciertas preguntas escritas en caracteres chinos, el hombre en el cuarto, con la ayuda del manual, puede devolverle al interrogador respuestas que hacen al interrogador pensar que el hombre en el cuarto sabe chino. Pero, como se ha especificado, el hombre en el cuarto no sabe, no entiende, chino; sabe reconocer figuras y él simplemente substituye unos dibujos que reconoce por su figura por otros según especifica el manual. Así que el hombre en el cuarto logra pasar el test de Turing sin entender el idioma. Ahora bien, si se toma el manual como representación de un programa algorítmico (de computadora) para contestar preguntas suministradas en chino y se considera al hombre que maneja el manual mecánicamente como una representación de la computadora misma, entonces tenemos un ejemplo vívido de la posibilidad de que se puede pasar el test de Turing sin el debido entendimiento. Según argumenta Searle, la manipulación de símbolos no es equivalente a inteligencia consciente (Searle, 1990, pp. 27-28).

Desde que el cuarto chino fue presentado por Searle han habido muchos argumentos en favor, y aún más en contra, del mismo, pues inclusive cuando apareció por primera vez fue acompañado de veintiséis comentarios (Searle, 1980). Resumir todos los argumentos en favor y en contra sería tema para otro artículo.[5] Mencionaré solamente uno de los más importantes, que es el factor de tiempo. Dado que el hombre en el cuarto chino no entiende chino y dado que tendría que preparar sus contestaciones a las preguntas en chino

[5] Para un resumen breve de la situación argumentativa en la actualidad, vea el artículo de Searle y el de Churchland y Churchland en la serie "Artificial Inteligence: A Debate" que aparece en el Vol. 262, Núm. 1 de la revista *Scientific American* (January 1990), pp. 25-37.

por medio de una comparación de las figuras suministradas con las figuras que aparecen en el manual y la reproducción de las figuras que le indica el manual, el tiempo que utilizaría para contestar preguntas sería mucho más extenso que el tiempo que necesitaría un conocedor corriente del idioma para producir semejantes respuestas. Dada la lentitud en recibir contestaciones, el interrogador dudaría que está hablando con una persona (que entiende chino), y el que está en el cuarto chino (en este caso el hombre que no conoce chino) no lograría pasar la prueba Turing.

Sin embargo, la cuestión de tiempo es un arma de doble filo. Como señalan Churchland y Churchland (1990, p. 33), ya para finales de la década de los 70 y principios de la década de los 80 se había notado que el tiempo requerido para la simulación de procesos cognoscitivos —como, por ejemplo, el reconocimiento de objetos en el sistema visual— era mucho mayor que el tiempo de sistemas reales, y esto es así a pesar de que la propagación de señales en las computadores es aproximadamente un millón de veces más rápido que la propagación de señales en los cerebros (eventos en computadores electrónicas ocurren en el orden de nano- —ó 10^{-9}— segundos mientras que eventos en las neuronas ocurren en el orden de mili- —ó 10^{-3}— segundos (Churchland y Sejnowski, 1992, pp. 8-9). Además se había notado que la simulación de comportamiento realista requiere acceso a una base muy amplia de conocimiento básico de trasfondo ("background information") lo cual resulta no solamente en el problema de tener que establecer esa base de conocimiento sino también en el problema mucho más intransigente de cómo considerar en tiempo realista solamente la parte relevante de tal base (Churchland y Churchland, 1990, p. 33; Glymour, 1992, pp. 351-352).

Observaciones de esta índole han impulsado a algunos defensores de la IA tradicional a modificar sus programas de investigación en busca de alternativas a la arquitectura funcional tradicional (o arquitectura "von Neumann") de las máquinas que manipulan símbolos. Para distinguir estos nuevos enfoques de IA, se tiende a denominar "clásico" al enfoque tradicional. Una de las alternativas que actualmente está de moda se llama Conexionismo o PDP ("parallel distributed processing"), y toma su inspiración de los estudios neurológicos. Siguiendo una especie de ingeniería a la inversa ("reverse engineering")[6] del cerebro, Conexionismo intenta acercar

[6] Práctica común en la industria según la cual se desmonta un producto o aparato nuevo (terminado) para descubrir su funcionamiento y poder duplicarlo.

la simulación cognoscitiva a los procesos neurales: el funcionamiento paralelo (versus el funcionamiento en serie de las computadores tradicionales) de una multitud de unidades simples. Dennett (1987, pp. 229-230) lo resume de la siguiente forma:

1. Procesamiento y memoria "distribuida" de tal forma que las unidades juegan múltiples roles drásticamente equívocas, y la disambiguación ocurre solamente de forma global;

2. no hay un control central sino un sistema parcialmente anárquico de elementos parcialmente competitivos;

3. no hay intercambio complejo de mensajes entre unidades;

4. hay una dependencia en la propiedades estadísticas de conjuntos para lograr efectos;

5. la construcción y destrucción de trayectos o soluciones parciales (al azar y posiblemente ineficientemente), hasta que después de un tiempo el sistema se acomoda (sin necesariamente haber logrado la contestación "correcta").

Dado estos nuevos giros, cabe preguntar: ¿Es o no es el funcionamiento de la mente humana análogo al funcionamiento de una computadora? En cierto sentido lo es, puesto que el cerebro computa funciones. Como afirma Dennett (1991, p. 269): "En el corazón de aun el sistema más volátil del reconocimiento de patrones (sea o no sea Conexionista) reside un motor tipo von Neumann . . . computando una función computable". Sin embargo, como se ha discutido, aun defensores de la IA fuerte han cuestionado si la mente es **exclusivamente** una computadora en el sentido de una máquina digital tipo von Neumann o Turing con arquitectura en serie. Precisar qué tipo de computadora es la mente sigue siendo proyecto de investigación para los científicos cognoscitivos, al igual que sigue como proyecto de investigación el intento de construir una inteligencia artificial que simule la mente humana en cuanto a sus funciones fundamentales. Sólo la experiencia y el tiempo nos dirá si somos capaces de producir tal inteligencia artificial. Y si se logra, ¿restaría del sentido de misterio y reverencia que tiende a asociarse con la existencia de la conciencia humana? ¡Todo lo contrario! El entendimiento no anula sino profundiza el sentido de reverencia.

REFERENCIAS[7]

(1990). "Artificial Intelligence: A Debate", en *Scientific American* 262, no. 1, pp. 25-37. (Contiene los artículos abajo mencionados: Searle 1990, y Churchland & Churchland, 1990).

Aspray, William. (1980). *From Mathematical Constructivity to Computer Science: Turing, Newmann, and the Origins of Computer Science in Mathematical Logic*. Ann Arbor, Mich.: University Microfilms International.

Churchland, Paul M. y Patricia Smith Churchland. (1990) "Could a Machine Think?", *Scientific American* 262, no. 1, pp. 32-37.

Churchland, Patricia S. y Terrence J. Sejnowski. (1992). *The Computational Brain*. Cambridge, Mass.: MIT Press.

Dennett, Daniel C. (1987). *The Intentional Stance*. Cambridge, Mass.: MIT Press.

______(1991). *Consciousness Explained*. Boston: Little, Brown & Co.

Goldstine, Herman H. (1972/1993). *The Computer: from Pascal to von Neumann*. Princeton, Princeton University Press.

Glymour, Clark. (1992). *Thinking Things Through*. Cambridge, Mass.: MIT Press.

Hermes, Hans. (1984). *Introducción a la teoría de la computabilidad: Algoritmos y Máquinas*. Trad. de Manuel Garrido y Aránzazu Martín Santos. Madrid: Tecnos.

Penrose, Roger. (1989). *The Emperor's New Mind: Concerning Computers, Minds, and the Laws of Physics*. Oxford: Oxford University Press.

Searle, John R. (1980). "Minds, Brains, and Programs", *Behavioral and Brain Sciences* 3, no. 3, pp. 417-458.

______ (1990). "Is the Brain's Mind a Computer Program? *Scientific American* 262, no. 1, pp. 26-31.

Turing, A.M. (1985) "¿Puede pensar una máquina?" Trad. de Manuel Garrido y Amador Antón, en A.M. Turing, H. Putnam y D. Davidson, *Mentes y Máquinas*. Madrid: Tecnos, pp. 16-60. (Original: A.M. Turing. (1950). "Computing Machinery and Intelligence", *Mind* 59, no. 236: 433-460.)

---

[7] Dado el creciente número de obras en el área de computación/inteligencia artificial, sólo se ha hecho referencia a obras específicamente mencionadas o citadas en el texto del artículo.

Ursua Lezaun, Nicanor. (199O). "Cerebro, Máquina y Algoritmo", en *El Nuevo Mundo de la Filosofía de la Tecnología*, eds. Carl Mitcham *et al.* University Park, PA: STS Press.

Halley D. Sánchez
Departamento de Humanidades
Universidad de Puerto Rico
Mayagüez, Puerto Rico 00681