

## ALAN MATHISON TURING

Coromoto León Hernández y Casiano Rodríguez León



Alan Turing

Al finalizar sus estudios de Historia y Literatura en la Universidad de Oxford, Julius Turing ingresó en el *Indian Civil Service*, el cuerpo de funcionarios con el que el Imperio Británico administraba la India. Después de diez años de trabajo en Madras, al sur del país, volvió de vacaciones a Inglaterra con la intención de buscar esposa. En el barco en el que viajaba conoció a Ethel Stoney con la que se casaría poco después. Julius y Ethel tuvieron dos hijos: John y Alan. Alan Turing fue concebido en la India pero nació en Paddington, cerca de Londres, el 23 de junio de 1912. Su madre regresa a la India y el niño Alan permanece en Inglaterra con familiares e interno en colegios. Su infancia se caracteriza por su gran dificultad para comunicarse y su gusto por lo *abstracto*. Estaba fascinado por las notaciones simbólicas y uno de sus juegos preferidos era descifrar los números de serie de las farolas de las calles.

El Turing adolescente era torpe en caligrafía, literatura y latín y sentía pasión por las *ciencias*. Se adaptaba bastante mal al sistema elitista de las *public school* británicas (curiosamente los ingleses denominan *public school* a las escuelas privadas). En 1926 ingresa en la Sherborne School donde le apodan con el nombre de *Math Brain* pues empleaba su tiempo en la lectura de obras de matemáticas de alto nivel. Llevaba un cuaderno de notas sobre teoría de la relatividad y desatendía los estudios del programa oficial. Incluso en matemáticas no obtiene sino resultados mediocres. Es entonces cuando conoce a Christopher Morcom, su primer y platónico amor. Morcom, un año mayor que Turing compartía con él su interés por las matemáticas y las ciencias. Cuando Morcom era pequeño tomó leche de vaca infectada con tuberculosis bobina; desde entonces su vida estuvo en constante peligro. Finalmente murió en febrero de 1930 y Alan quedó sumido en una larga crisis. Sin embargo, consigue salir de su bache emocional superando todas las asignaturas y consiguiendo una beca para el King's College de la Universidad de Cambridge donde ingresó en 1931.



Christopher Morcom

Cambridge, era en esa época uno de los centros de investigación matemática más importantes del mundo. El King's College se había convertido, en una Inglaterra todavía muy puritana, en el foco de irradiación de un pensamiento *liberal-libertario*, del que Turing absorbe todas las ideas. En esa época aceptó su condición de homosexual. Trabajaba mucho y practicaba deporte, sobre todo remar y correr. Lee la obra de Von Neumann sobre mecánica cuántica. Finalmente se gradúa en 1934.

En marzo de 1935, con sólo veintidós años, Alan fue elegido *fellow* del King's College, excelente noticia que significaba un sueldo de 300 libras al año. En la primavera de este año, asistió como alumno al seminario que Maxwell Newman impartió sobre *Fundamentos de las Matemáticas* en la Universidad de Cambridge. El seminario estaba basado en los trabajos que se presentaron en el congreso internacional de Matemáticas de 1929, en el que el matemático alemán David Hilbert desarrolló algunos de los *diecisiete problemas abiertos* que había formulado al comienzo del siglo XX. En concreto, abordó tres cuestiones:

- Uno: ¿son “completas” las matemáticas? Esto es, ¿puede cualquier “sentencia del lenguaje de las matemáticas” ser demostrada o rebatida?

- Dos: ¿son “consistentes”? ¿no ocurrirá la desgracia de que alguna sentencia sea demostrable y también lo sea su negación?, ¿se puede demostrar, dentro del propio marco de las matemáticas, que eso no ocurre?

- Tres: ¿son “computables”? ¿existe un “procedimiento mecánico” mediante el cual, teniendo como entrada la aserción que se va a demostrar, se produzca como salida, tras un número finito de pasos, la respuesta correcta acerca de su veracidad, falsedad o, si tal fuera el caso, su independencia del marco axiomático?

46

Las clases de Newman terminaron con la demostración del teorema de Gödel. Kurt Gödel había demostrado no sólo que la aritmética es incompleta (uno), sino que su consistencia (dos) no puede ser probada dentro de su propio marco axiomático. Así Newman condujo a Alan a lo que entonces eran las fronteras del conocimiento, proporcionándole las armas lógicas para sacar adelante el trabajo que aparecerá en *On Computable Numbers*. Los trabajos de Turing provocaron que, desde un punto de vista lógico-matemático se planteara la cuestión de la computabilidad (tres). Su teoría demostraba que para ciertos problemas, a pesar de que estén perfectamente bien planteados, puede no existir ningún procedimiento que, en un número finito de operaciones, los resuelva. Esto es lo que se denomina un problema *no computable* (como diría el robot en *Perdidos en el espacio*) o indecidibles según la terminología de Gödel. En los párrafos siguientes se expone con terminología actual cuales fueron las ideas de Turing.

Cualquier conocimiento se transmite a través de un *lenguaje*. Las frases que componen un lenguaje como el español, el lenguaje de la Lógica o un lenguaje de programación como Java no son más que la concatenación de los elementos de un alfabeto finito. Matemáticamente: si fijamos un alfabeto

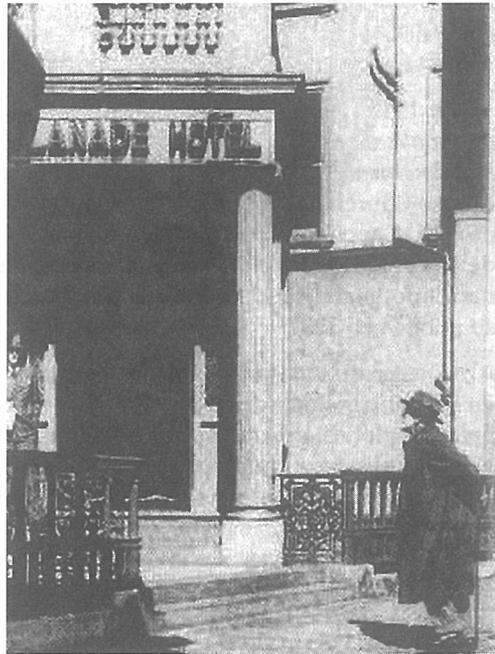
$$\Sigma = \{0, \dots, 9, a, b, \dots\}$$

suficientemente rico en símbolos, podemos considerar el conjunto *cierre de Kleene* de  $\Sigma$ , denotado por  $\Sigma^*$  y formado por todas las concatenaciones posibles de elementos de  $\Sigma$ :

$$\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup \dots \cup \Sigma^n \cup \dots$$

Aquí  $\Sigma^n$  denota el producto cartesiano de  $\Sigma$  por sí mismo  $n$  veces. Si entendemos el *español* como el conjunto de frases (discursos, textos) que lo componen, entonces cabe entender que el Español es un subconjunto de  $\Sigma^*$ . Es claro que lo mismo sucede con otros lenguajes como el *inglés*, la *Lógica* y todos los *Lenguajes de programación*. En  $\Sigma^*$  conviven todas las novelas y tratados escritos y que se escribirán, todos los programas que fueron y serán, con todas las combinaciones absurdas de símbolos. Es una *Biblioteca de Babel* como la que describía Jorge Luis Borges en su famoso cuento.

Definiremos *lenguaje* en un sentido amplio, como cualquier subconjunto de  $\Sigma^*$ , y *lenguaje computable* es aquel lenguaje  $L \subseteq \Sigma^*$  para el cuál existe un programa  $P$  que discierne si una frase  $x \in \Sigma^*$  está o no en  $L$ . La palabra *programa* es usada aquí en su sentido habitual. Con una excepción: si bien suponemos que durante la ejecución del programa sólo se hace uso de una cantidad de memoria finita, suponemos que nuestro computador dispone de una capacidad potencial de almacenamiento infinita (esto es, admitimos que siempre podemos ir a la tienda de la esquina a comprar más memoria si nos hace falta). Nótese que la definición de lenguaje computable no entra en veleidades semánticas, es simplista y no pretende siquiera que el programa que computa  $L$  entienda o no el significado de las frases  $x$  de  $L$ ; simplemente pretende que nos diga si  $x$  es o no un elemento de  $L$ . Es cierto que los lenguajes para programar computadoras como FORTRAN, Pascal, Cobol, C, Java o Matemática son “lenguajes computables”. La prueba: para ellos existen los



Alan Turing nació en una clínica del distrito de Paddington en Londres llamada Warrington Lodge. El edificio estuvo dedicado a este fin hasta 1935 en que pasó a ser el Hotel Esplanade, utilizado por muchos refugiados de Europa continental, entre ellos Sigmund Freud.

En la foto se ve a Freud a las puertas del hotel en 1938 después de haber abandonado Viena.

En la actualidad es el Hotel Colonnade.

correspondientes programas, denominados “compiladores” que, cuando se les da como entrada un programa mal escrito se quejan con un *syntax error*, pero si se hace bien ejecutarán las acciones que se solicitó.

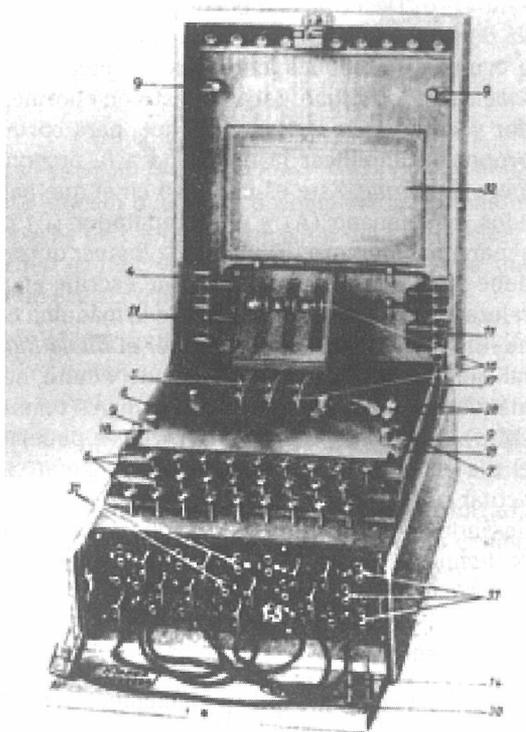
¿Es cualquier lenguaje, cualquier subconjunto de  $\Sigma^*$  computable? ¿Puede un lenguaje como el *español*, el *inglés* o los *Teoremas Lógicos Ciertos* ser “reconocido” por un computador? Puede obtenerse una primera respuesta utilizando la teoría de cardinales transfinitos de Cantor. En dicha teoría, un conjunto para el que existe una correspondencia biyectiva con el conjunto de los números naturales se dice *numerable*. Se puede probar que  $\Sigma^*$  es un conjunto numerable, ya que es la unión infinito-numerable de conjuntos finitos  $\Sigma^n$ . Puesto que cualquier programa de ordenador es un elemento de  $\Sigma^*$ , se sigue que el número de programas que se hayan escrito o jamás puedan escribirse es numerable. Sin embargo, el número de lenguajes sobre el alfabeto  $S$  no es numerable, pues, tal como Cantor demostró, no se puede encontrar una correspondencia biyectiva entre el conjunto  $2^{\Sigma^*}$  de los subconjuntos de  $\Sigma^*$  y el propio conjunto  $\Sigma^*$ . Por tanto, deben existir lenguajes que no sean computables. La demostración de Turing sobre la existencia de lenguajes y funciones no computables es constructiva. Un corolario famoso de su teoría es que el *problema de la parada* es no computable. El *problema de la parada* está definido por el lenguaje de los pares (*programa, entrada*), tales que el *programa*, al ser ejecutado sobre la *entrada*, termina, esto es, no cae en ninguna clase de bucle infinito.

46 En 1936, Turing intenta publicar su trabajo *On Computable Numbers: with an application to the halting problem* en el libro de actas de la *London Mathematical Society*. Conseguirlo no fue fácil. La razón de la negativa fue que Alonzo Church había publicado en la *American Journal of Mathematics* su tesis de que la noción de función recursiva no es sino la formalización matemática de la noción de función humanamente computable. Turing revisa su trabajo haciendo una mención a la tesis de Church y finalmente su artículo revisado se publica en 1937.

Como resultado de sus discusiones con Church, en 1936, Turing se traslada a la Universidad de Princeton (New Jersey, Estados Unidos), para realizar investigaciones conjuntas. En Princeton trabaja en la demostración de la equivalencia entre su concepto de computabilidad y el de Church. Teoremas de equivalencia como estos son los que, confirmados una y otra vez para otros modelos de computación propuestos posteriormente, justifican la conocida *tesis de Church-Turing*. Dicha tesis propone que, supuesta una capacidad de almacenamiento ilimitada, cualquiera que sea la forma en la que un computador se construya, su capacidad potencial de cómputo no varía (aunque si puede variar su velocidad). Introduce la idea de máquina oráculo (una máquina que es capaz de dar la respuesta a una cuestión indecidible). Esto es, supuesta computable una cierta función, ¿qué nuevas funciones podemos computar? Turing recibe su doctorado por Princeton en 1938 con el trabajo titulado *Systems of Logic Based on Ordinals* y regresa a Inglaterra.

En 3 de septiembre de 1939, Inglaterra entró en la Segunda Guerra Mundial. El departamento británico de análisis criptográfico pidió a Turing que colaborara con ellos. Los alemanes usaban una máquina llamada *Enigma* para cifrar

sus mensajes. El proyecto aprovechó las ideas de un grupo de matemáticos polacos, que habían desarrollado una máquina denominada *La bomba* capaz de descifrar ciertos códigos particulares de Enigma. Turing logró generalizar las instrucciones de la Bomba y acelerar el proceso de descifrado en un factor de veintiséis. A finales de 1939 eran capaces de desentrañar los mensajes de la aviación alemana, en 1941 descifraban los de la marina. Sin embargo, en febrero de 1942 los alemanes cambiaron el sistema de encriptado. En un esfuerzo coordinado con los americanos se logró de nuevo en 1943 el desciframiento, que se mantuvo hasta el final de la guerra. En 1945 Turing recibió la Orden del Imperio Británico como reconocimiento a su labor.



*La máquina Enigma.*

En 1946 Turing es nombrado investigador principal de un proyecto del *National Physical Laboratory* de Londres con el que se pretende construir un computador digital. Turing diseña un computador universal con programa almacenado en memoria, el ACE (*Automatic Computing Engine*) y establece las bases de su programación. En octubre de 1947 vuelve por un año a Cambridge donde estudia neurología y fisiología. Al año siguiente, se traslada a la Universidad de Manchester aceptando el puesto que le ofrece Newman. En 1950 publica *Computing machinery and intelligence in Mind*. En ese trabajo, Turing casi escribió el guión de la película *Blade Runner* al intentar contestar a la cuestión *¿Puede pensar una máquina?* (*Cosa difícil de definir la inteligencia es, que diría el pequeño Yoda*).

¿Recuerdan la primera escena de *Blade Runner*? En algún edificio de la galáctica ciudad de Los Ángeles, un tipo con aspecto de policía o psiquiatra cuestiona a un hombre sudoroso y preocupado. Las preguntas parecen sacadas de uno de esos test psicológicos que aparecen en los semanarios. Algo así como, "Está usted en el desierto caminando por la arena, cuando ve un galápagos que se arrastra hasta usted. Usted se agacha, lo pone patas arriba y lo

deja cociéndose al sol. El galápagos trata de darse la vuelta. ¿Por qué no le ayuda?” o “Describe con palabras sencillas, sólo las cosas buenas que recuerda de su madre” El hombre está cada vez más incómodo. Ante alguna de las preguntas, se bloquea. Entonces el policía, el *Blade Runner*, sabe. Y el hombre sabe que el otro sabe. Saca un pistolón enorme, descerraja dos tiros al inquisidor y salta por la ventana. Así que, para cortar por el nudo gordiano, Turing proponía simplificar la respuesta a la pregunta planteada con un juego: *El juego de Turing*. Este es un juego en el que participan tres jugadores. Dos de ellos, un humano (A) y un computador (C) permanecen en una habitación separada. El tercero (B, ¿un *Blade Runner* quizá?), ignorante de quién es quién, debe mediante las cuestiones que escribe en una terminal averiguar la naturaleza de cada uno. El objetivo de la máquina es, por tanto, suplantar al humano. El del humano A es convencer al *Blade Runner* B de su condición de mortal. Si un *Blade Runner* B, inteligente y culto, no es capaz de distinguir entre la máquina y el interlocutor humano A (también seleccionado inteligente y culto), entonces —arriesgaba Turing— podemos decir que C piensa. ¿Llegará el día en el que una máquina pueda ganar este juego? Ningún computador actual es capaz de tal proeza. Si llega ese momento, ¿podremos seguir considerándola una máquina? Las respuestas a esta pregunta han evolucionado con el tiempo. Más que en el campo matemático-lógico, las mayores dificultades que presenta la *simulación* de la conducta humana por un ordenador parecen residir en los problemas de percepción, en los de relación con el entorno y en las emociones.

46

En 1951, Turing fue elegido miembro de la *Royal Society* de Londres. A finales de este año, Turing llevó a su casa a Arnold, un joven de diecinueve años que había encontrado en un lugar frecuentado por homosexuales en Oxford Street, Manchester. Posteriormente, un amigo de Arnold entró en casa de Turing y le robó. Turing denunció el robo. La policía, al contrario de lo que pudiera esperarse, concentró su investigación en la vida privada de Turing. En febrero de 1952, es arrestado y acusado de conducta indecente. Le ofrecen cambiar la pena de cárcel por un tratamiento hormonal. El tratamiento arruinó su salud. Tuvo que abandonar los trabajos en criptografía que realizaba para el gobierno (estos habían cobrado gran importancia con la llegada de la Guerra Fría). La policía de la seguridad del estado le vigilaba constantemente, llegando a registrar su casa y a investigar a los colegas extranjeros que recibía en la universidad. El 7 de junio de 1954, come una manzana inyectada de cianuro. La mujer de la limpieza lo encontró muerto en la cama el día siguiente. Tenía 42 años.