Tema 2: Máquinas de Turing

1. Antecedentes históricos.

En el siglo XVII, Leibniz planteó la búsqueda de un procedimiento efectivo que permitiera resolver cualquier problema matemático; esto es, se intentaba buscar un método general que nos diera la solución a un problema arbitario. A principios del siglo XX, David Hilbert resumió la búsqueda de este método general en una conjetura que afirmaba que las Matemáticas en su conjunto podían axiomatizarse. Intentó buscar un algoritmo que permitiese determinar la validez o no de cualquier proposición matemática. Sin embargo, ante los infructuoso intentos de construcción, se empezó a pensar que quizás el procedimiento buscado no existía, es decir, que era posible la existencia de problemas para los cuales no era factible crear un algortimo resolutivo. Siguiendo esta nueva línea, en 1931 el lógico austriaco Kurt Gödel publicó un artículo en el que se recogía su famoso teorema de incompletitud, que probaba la no existencia del procedimiento efectivo buscado. Así, Gödell construyó una fórmula en el cálculo de prediciados aplicada a los enteros cuya definición establecía que no podía ser probada su validez o falsedad dentro del sistema lógico. Esto constituyó un gran avance para la lógica matemática, pues permitió formalizar la noción intuitiva de procedimiento efectivo y abandonar los intentos de construcción del algoritmo general que se habían realizado hasta entonces.

En este marco surgieron las máquinas de Turing que eran un mecanismo que permitían resolver problemas concretos. Simultaneamente, Church, Kleene y Post presentaron otros modelos que teníanuna capacidad de computación semejante al modelo de Turing, aunque este último ha sido el más conocido e investigado.

Para diseñar su máquina, Turing se basó en las calculadoras que en aquella época existían. Gracias a ellas, se dió a la computabilidad un significado más concreto y preciso y se describió lo que hoy conocemos como **algoritmo**: una serie de instrucciones que al seguirlas resuelven el problema para el que han sido creado. De manera rápida, podemos visualizarlas como una computadora idealizada en las que las operaciones elementales se descomponen hasta el límite. De hecho, se pueden considerar el dispositivo teórico precusor de los computadores actuales, con dos salvedades: la cantidad de memoria que dispone es infinita y las operaciones más simples se descomponen lo más posible. Precisamente, esta última cuestión es la que marca las diferencias más claras entre los ordenadores y el dispositivo teórico creado por Turing: entre otros, los objetivos que presentan los computadores actuales es el de procesamiento de la información de la manera más rápida posible y la realización de complejas operaciones matemáticas con el menor coste operativo y con la simplicidad de la máquina de Turing el coste operativo que se tiene es alto, si entendemos por tal el número de pasos que se deben dar para conseguir la solución deseada.

Por último, para finalizar este párrafo debemos mencionar que ideas desarrolladas por Turing en su concepción de las máquinas de Turing son en la actualidad empleadas en programación. Así, la composición en paralelo de máuinas que presentó Turing es similar a la que hoy se está empleando en programación. Por otro lado, sería erroneo pensar que la única finalidad de las máuinas de Turing se centra en ser dispositivos teóricos cuyas ideas sirvieron para diseñar computadores. También han sido empleados en la lógica matemática par demostrar resultados como el Teorema de Markov-Post, relacionadolas con los subconjuntos recursivamente enumerables.

2. Descripción de una máquina de Turing.

Intuitivamente, podemos interpretar una máquina de Turing como cabeza lectora-inscriptora a través de la cual pasa una cinta que consta de una serie de casillas contiguas y es tan larga a derecha y a izaquierda como se desee. La cabeza lectora-inscriptora es capaz de imprimir un número finito de símbolos $s_0,\ s_1,\dots,\ s_m$ y de estar en un número finito de estados $e_0,\ e_1,\dots,\ e_n$. En un momento dado, la cabeza se encuentra sobre una casilla determinada que contiene un único símbolo s_j en un estado e_i y dependiendo de las instrucciones que posea realizará una de las siguientes acciones: moverse una casilla a la derecha, moverse una casilla a la izquierda, cambiar el símbolo s_j por s_k ó parar. Opcionalmente, en las tres primeras posibilidades variará su estado interno y continuará el proceso descrito.

Procedemos a dar la definición formal de máuina de Turing. Salvo que se indique lo contrario, trabajaremos con $S = \{s_0, s_1, \dots, s_m\}$ conjunto finito cuyos elementos denominamos símbolos y con $E = \{e_0, \dots, e_n\}$ conjunto finito cuyos elementos llamamos estados.

Necesitamos introducir previamente la noción de cuádruple:

Definición. Un **cuádruple** es una 4-tupla de uno de los tres tipos siguientes:

i) $e_i s_j s_k e_l$

ii) $e_i s_j D e_l$

iii) $e_i s_j I e_l$,

donde $e_i,\ e_l \in E,\ s_j,\ s_k \in S$ y D e I denotan derecha e izquierda, respectivamente.

Por abuso del lenguaje, hemos omitido el símbolo (, , ,), entendiendo que dos expresiones del tipo anterior son iguales si y sólo si tienen iguales sus componentes. Cada uno de estos cuádruples corresponden a un tipo de movimiento de la descripción intuitiva dada anteriormente. Esto es:

i) $e_i s_j s_k e_l$ se interpreta de la manera siguiente: al examinar el símbolo s_j en el estado e_i, borra el s_j, imprime s_k y entra en el estado e_l.

ii) $e_i s_j D e_l$ significa: cuando examines el símbolo s_j en el estado e_i, mueveté una casilla hacia la derecha y pasa al estado e_l.

iii) $e_i s_j I e_l$ significa: cuando examines el símbolo s_j en el estado e_i, mueveté una casilla hacia la izquierda y pasa al estado e_l.

Dos cuádruples serán distintos cuando difieran en una de las componentes del mismo.

Definición. Sean $S = \{s_0, \ldots, s_m\}$, $E = \{e_0, \ldots, e_n\}$ y $\mathfrak{C}$ tres conjuntos finitos no vacíos cuyos elementos son símbolos, estados y cuádruples, respectivamente. Diremos que $\mathfrak{T} = (S, E, \mathfrak{C})$ es una **máquina de Turing** si $\mathfrak{C}$ satisface la propiedad siguiente: no existen dos cuádruples distintos en $\mathfrak{C}$ que comiencen por $e_i s_j$ para algunos $i \in \{0, \ldots, n\}$ y $j \in \{0, \ldots, m\}$.

Equivalentemente, una máquina de Turing consiste en dar una correspondencia

$$\begin{array}{rccc} f: & E \times S & \longrightarrow & (S \cup \{I, D\}) \times E \\ & (e_i, s_j) & \longmapsto & f(e_i, s_j) \end{array}$$

que cumple que a cada (e_i, s_j) le corresponde a lo sumo un $f(e_i, s_j)$.

Observemos que:

i) El conjunto imagen de esta correspondencia consiste de elementos del tipo (s_k, e_l), (D, e_l) ó (I, e_l).

ii) La correspondencia anterior no es en general una aplicación en el sentido que conocemos, puesto que no todos los elementos de $E \times S$ tienen una imagen por f.

La condición de que no existan dos cuádruples que comiencen con el mismo par de elementos de $E \times S$, esto es, para cada $(e_i,s_j) \in E \times S$ se verifica $|\{f(e_i,s_j)\}| \leq 1$, supone que en cada instante el movimiento a realizar por la máquina está totalmente determinado, no cabe ambigüedad sobre lo que tiene que hacer.

La máquina se parará si se encuentra en un estado e_i examinando un símbolo s_j y no existe ningún cuádruple que comience por $e_i s_j$.

Ejemplo. Se consideran los conjuntos $S = \{s_0, s_1, s_2, s_3\}$ y $E = \{e_0, e_1, e_2\}$. Estos conjuntos junto con los siguientes cuádruples:

$$\mathfrak{C} = \{e_0 s_0 D e_1, e_1 s_1 s_2 e_2, e_0 s_1 I e_2, e_2 s_0 I e_2, e_0 s_2 I e_1, e_2 s_3 s_1 e_0\}$$

constituyen una máquina de Turing.

Interpretando la máquina anterior como una correspondencia tenemos:

$$
\begin{array}{rccc}
f: & E \times S & \longrightarrow & (S \cup \{I, D\}) \times E \\
 & (e_0, s_0) & \longmapsto & (D, e_1) \\
 & (e_0, s_1) & \longmapsto & (I, e_2) \\
 & (e_0, s_2) & \longmapsto & (I, e_1) \\
 & (e_1, s_1) & \longmapsto & (s_2, e_2) \\
 & (e_2, s_0) & \longmapsto & (I, e_2) \\
 & (e_2, s_3) & \longmapsto & (s_1, e_0)
\end{array}
$$

También podemos visualizar las acciones de una máquina de Turing mediante una tabla. Así, los cuádruples del ejemplo anterior vienen dados por:

	s_0	s_1	s_2	s_3
e_0	De_1	Ie_2	Ie_1	
e_1		s_2e_2		
e_2	Ie_2			s_1e_0

Para elaborar este cuadro hemos utilizado el siguiente criterio: Cuando la máquina se encuentre en el estado e_i examinando una casilla que lleve el símbolo s_j, en la casilla $(i,\ j)$ correspondiente aparece el movimiento a realizar. Si no hay nada, significa que la máquina se para.

Del hecho de que para cada par (e_i, s_j) pueda haber a lo sumo un cuádruple que comience por él, se sigue que podemos contar el número de máquinas de Turing distintas que se pueden dar con un mismo conjunto de símbolos y estados, entendiendo que dos máquinas serán diferentes si presentan, al menos, un cuádruple distinto.

Teorema 2.1. *El número de máquinas de Turing diferentes que se pueden crear con m símbolos y n estados es $(n(m+2)+1)^{nm} - 1$,*

3. Diseño de máquinas de Turing con objetivos prefijados.

A continuación vamos a diseñar máquinas de Turing que realicen tareas concretas. En lo que sigue, salvo que se indique lo contrario, denotaremos una casilla vacía en la cinta de la máquina por el símbolo s_0. Llamaremos **estado de partida** de una máquina de Turing al estado en el que se encuentra el dispositivo cuando comienza a actuar.

Definición. Sea $\mathfrak{T} = (S, E, \mathfrak{C})$ una máquina de Turing. Se llama **descripción instantánea de** $\mathfrak{T}$ a una palabra $\alpha = Pe_is_jP'$, donde $P, P' \in \Omega_S$, $e_i \in E$ y $s_j \in S$.

Intuitivamente, una descripción instantánea debe entenderse de la manera siguiente: "los símbolos de la cinta son las letras que aparecen en $P\ s_j$ y P' (escritos éstos en celdas contiguas y casillas en blanco en el resto) y la cabeza lectora-inscriptora se encuentra en el estado e_i examinando la casilla que contiene s_j".

Por ejemplo, la descripción instantánea $\alpha = s_2e_3s_1s_5s_7s_2$ la interpretamos: "los símbolos de la cinta son $s_2s_1s_5s_7s_2$ y la máquina se encuentra en el estado e_3 examinando una casilla que contiene a s_1".

Definición. Sea $\mathfrak{T} = (S, E, \mathfrak{C})$ una máquina de Turing y $\alpha = Pe_is_jP'$ una descripción instantánea de $\mathfrak{T}$. Decimos que α es **terminal** si no existe un cuádruple en $\mathfrak{C}$ que comience por e_is_j.

Definición. Sea $\mathfrak{T} = (S, E, \mathfrak{C})$ una máquina de Turing y α y β dos descripciones instantáneas. Escribiremos $\alpha \to \beta$ siempre que existan $P, P' \in \Omega_S$ de manera que se verifique una de las siguientes condiciones:

i) $\begin{array}{rcl} \alpha & = & Pe_is_jP' \\ \beta & = & Pe_ls_kP' \end{array}$ y el cuádruple $e_is_js_ke_l \in \mathfrak{C}$

ii) $\begin{array}{rcl} \alpha & = & Pe_is_js_kP' \\ \beta & = & Ps_je_ls_kP' \end{array}$ y el cuádruple $e_is_jDe_l \in \mathfrak{C}$

iii) $\begin{array}{rcl} \alpha & = & Pe_is_j \\ \beta & = & Ps_je_ls_0 \end{array}$ y el cuádruple $e_is_jDe_l \in \mathfrak{C}$

iv) $\begin{array}{rcl} \alpha & = & Ps_ke_is_jP' \\ \beta & = & Pe_ls_ks_jP' \end{array}$ y el cuádruple $e_is_jIe_l \in \mathfrak{C}$

v) $\begin{array}{rcl} \alpha & = & e_is_jP' \\ \beta & = & e_ls_0s_jP' \end{array}$ y el cuádruple $e_is_jIe_l \in \mathfrak{C}$.

Diremos que $\alpha \to \beta$ es un **movimiento básico** de la máquina T.

Definición. Una **computación** de una máquina de Turing $\mathfrak{T} = (S, E, \mathfrak{C})$ es una sucesión finita de descripciones instantáneas $\alpha_1, \alpha_2, \ldots, \alpha_t$ tales que $\alpha_i \to \alpha_{i+1}$ para $i = 1, 2, \ldots, t-1$ y α_t es terminal.

Ejemplo. Consideramos la máquina $\mathfrak{T} = (S, E, \mathfrak{C})$ con $S = \{s_0, s_1, s_2\}$, $E = \{e_0, e_1, e_2, e_3\}$ y $\mathfrak{C} = \{e_0s_0Ie_0, e_0s_1De_1, e_0s_2De_1, e_1s_0De_1, e_1s_2s_1e_1, e_1s_1De_2, e_2s_0s_0e_3, e_2s_2s_2e_2\}$. Entonces, la sucesión de descripciones instantáneas $\alpha_1 = s_2e_1s_2s_1$, $\alpha_2 = s_2e_1s_1s_1$, $\alpha_3 = s_2s_1e_2s_1$ es una computación. En cambio, si partimos de $\alpha_1 = s_2e_1s_2s_2$, tenemos $\alpha_2 = s_2e_1s_1s_2$ y $\alpha_i = s_2s_1e_2s_2$ para $i \geq 3$, es una sucesión infinita de descripciones instantáneas tales que $\alpha_i \to \alpha_{i+1}$ que no es una computación.