



Lógica Superior: **Computabilidad e Incompletitud**

Profesor: Eduardo Barrio
UBA - Filosofía
2do cuatrimestre de 2015



Tesis de Turing-Church

Las nociones de *algoritmo* y de *computable* son intuitivas.

Algoritmos \Leftrightarrow máquina de Turing. (Tesis de Turing)

Algoritmos \Leftrightarrow Funciones recursivas (Tesis de Church)

Algoritmo para preparar correctamente un buen té en saquitos:

- Colocar el saquito en la taza ó tetera.
- Calentar el agua hasta el primer hervor (no debe ser sobrehervida) y verter sobre el saquito.
- Dejar reposar 3 minutos.



“Instrucciones para subir una escalera.” Julio Cortazar.

Nadie habrá dejado de observar que con frecuencia el suelo se pliega de manera tal que una parte sube en ángulo recto con el plano del suelo, y luego la parte siguiente se coloca paralela a este plano, para dar paso a una nueva perpendicular, conducta que se repite en espiral o en línea quebrada hasta alturas sumamente variables.

Agachándose y poniendo la mano izquierda en una de las partes verticales, y la derecha en la horizontal correspondiente, se está en posesión momentánea de un peldaño o escalón. Cada [uno](#) de estos peldaños, formados como se ve por dos elementos, se situó un tanto más arriba y adelante que el anterior, principio que da sentido a la escalera, ya que cualquiera otra combinación producirá formas quizá más bellas o pintorescas, pero incapaces de trasladar de una [planta](#) baja a un primer piso.

Las escaleras se suben de frente, pues hacia atrás o de costado resultan particularmente incómodas. La actitud natural consiste en mantenerse de pie, los brazos colgando sin esfuerzo, la cabeza erguida aunque no tanto que los ojos dejen de ver los peldaños inmediatamente superiores al que se pisa, y respirando lenta y regularmente. Para

subir una escalera se comienza por levantar esa parte del cuerpo situada a la derecha abajo, en el cuerpo o gamuza, y que salvo excepciones cabe exactamente en el escalón. Puesta en el primer peldaño, que para abreviar llamaremos pie, se recoge la parte equivalente de la izquierda (también llamada pie, pero no confundirse con el pie antes citado), y llevándola a la altura del pie, se le hace seguir hasta el segundo peldaño, con lo cual en éste descansará el pie, y en el primero descansará el pie. (Los primeros peldaños son los más difíciles, hasta adquirir la coordinación necesaria. La coincidencia de nombre entre el pie y el peldaño es una explicación. Cuidese especialmente de no levantar al mismo tiempo el pie y el peldaño).

Llegando en esta forma al segundo peldaño, basta repetir alternadamente los movimientos hasta el final de la escalera. Se sale de ella fácilmente, con un ligero golpe de talón que la fija en su sitio, del que no se moverá hasta el momento del descenso.





Funciones y computabilidad

¿Qué funciones son computables?

Definición informal:

Una función g es *efectivamente computable* si hay instrucciones explícitamente definidas y mecánicas para computar cada valor de g .

- Ignorando las limitaciones físicas de tiempo, velocidad y almacenamiento.





Funciones y computabilidad

¿Qué funciones son computables?

Pregunta 1:

¿Cómo ofrecer una noción precisa de computabilidad? [Turing- computability]

Pregunta 2:

¿Cuales son algunos ejemplos de funciones computables?

- Duplicar el número marcas en la cinta.
- Sumar
- Multiplicar



Funciones y computabilidad

¿Qué funciones son computables?

Pregunta 3:

¿Hay funciones no computables?

- **Si:** hay demasiadas funciones como para que todas sean turing computables.

- El conjunto de todos los subconjuntos de enteros positivos es no numerable.
- El conjunto de funciones Turing computables es enumerable.
- Por lo tanto, existen funciones que no son computables por una máquina de Turing

Pregunta 4:

Puede brindarse algún ejemplo de función no computable?

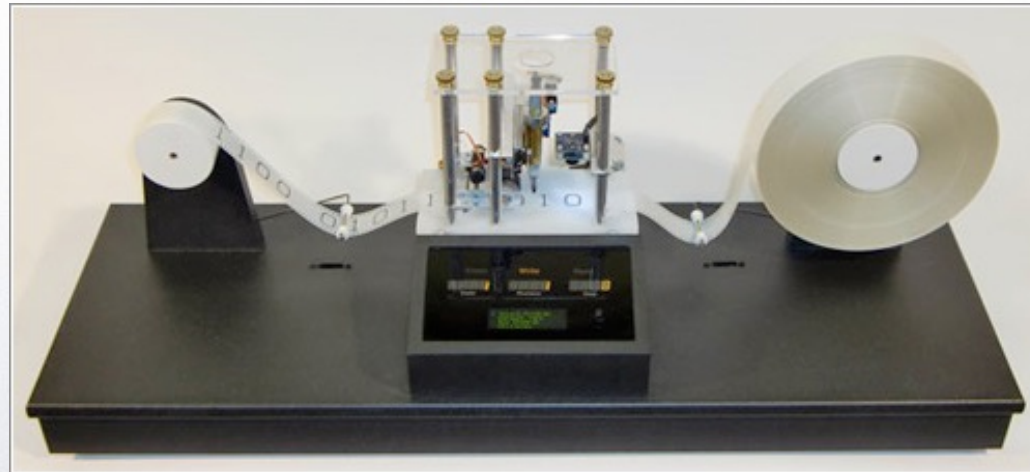
- Funciones diagonales: hay genuinas funciones d (de un argumento) que no son Turing computables.



Funciones y computabilidad

¿Qué funciones son computables?

Tesis de Turing: cualquier función efectivamente computable es Turing Computable.





Funciones y computabilidad

¿Qué funciones son computables?

Máquina de Turing: es un dispositivo que manipula símbolos sobre una cinta de acuerdo a una tabla de reglas. A pesar de su simplicidad, una máquina de Turing puede ser adaptada para simular la lógica de cualquier algoritmo de computador.



Video: Lego Turing Machine: <http://vimeo.com/44202270>



Máquina de Turing:

¿Qué es una máquina de Turing?

Componentes

Cinta: sin fin en ambas direcciones, dividida en cuadrados.

Símbolos:

1. - un conjunto finito S_0, S_1, \dots, S_n
2. - se usa S_0 (B o 0) para señalar celdas en blanco
3. - se usa 1 para S_1 .
4. - Sólo un símbolo es escrito en cada cuadrado

1. Acciones:

1. La máquina “ve” una celda y “lee” el símbolo impreso
2. La máquina está en uno de los finitamente muchos estados internos q_1, \dots, q_n
3. Condicionada al símbolo que lee y el estado en el que está, la máquina ejecuta una acción manifiesta y una encubierta.
4. Acciones manifiestas:
 - Parar la computación
 - Moverse una celda hacia la derecha
 - Moverse una celda hacia la izquierda
 - Escribe 0 en lugar de lo que hay
 - Escribe S_n en lugar de lo que hay.
5. Acciones encubiertas.
 - Asigna un nuevo estado interno.

A menos que sea establecido de otra forma, la máquina siempre comienza desde el estado q_1 .



Máquina de Turing:

Representando máquinas

1 - Tabla de máquina.

		S_0	S_1	(Símbolo leído)
Estado	q_1	S_1q_1	Lq_2	
Presente	q_2	S_1q_2	Lq_3	
	q_3	S_1q_3		

2- Conjunto de cuádruplos.

$q_1S_0S_1q_1$ $q_1S_1Lq_2$
 $q_2S_0S_1q_2$ $q_2S_1Lq_3$
 $q_1S_0S_1q_3$

3 - Diagrama (Flow Chart).

$S_0:S_1$	$S_0:S_1$	$S_0:S_1$
q_1	q_2	q_3



Máquina de Turing:

Representando máquinas

Máquina:

Escriba el símbolo S_1 tres veces.

La cinta tiene una cadena continua de 0s y 1s en todas partes. Comience en el 1 que esté más a la derecha, escriba un 1 adicional a la izquierda, vuelva a repetir dos veces el procedimiento y pare.



Máquina de Turing:

Notación monádica para representar $\langle 3,4 \rangle$

BBB111B1111BBBB

Definición de *función Turing computable*

1. - Se usan máquinas de Turing para definir esa clase de funciones
2. - Esas máquinas leen y escriben 0 (S_0 , B) o 1 (S_1)



Máquina de Turing:

Caso más simple: funciones de un argumento

Una máquina de Turing en un estado interno 1, leyendo el 1 en el bloque de 1s que esté más a la izquierda sobre una cinta en blanco se dice que está en una **posición estándar de inicio**.

La misma configuración (omitiendo el estado interior 1) es también la **posición estándar de final** (SFP).

Una máquina de Turing M define (computa) una función $f: P \rightarrow P$ (parcial) ssi
Comenzando en su configuración inicial (M está en q_1 leyendo el primer 1 en el bloque de 1s que esté más a la izquierda), para cada entero positivo a :
 $f(a) = b$, si la máquina **PARA** siguiendo sus instrucciones quedando en su configuración final.
 $f(a) =$ indefinida, si M no para en el estado estándar final o no para en absoluto



Máquina de Turing:

Caso más simple: funciones de un argumento

M define la función:

$$F(a) = a + 1.$$

Definición $f: P \rightarrow P$ es **Turing computable** ssi hay una **máquina de Turing** (que lee sólo 0s y 1s) que define f .