Arellano Granados Angel Mariano
218123444

Actividad 9 Máquinas De Turing - Teoría

1. Escriba un ensayo sobre el funcionamiento de las Máquinas de Turing y su uso en la actualidad

El inglés Allan Turing creo el modelo matemático de una maquina lógica llamada máquina de Turing que a pesar de ser muy simple contiene el poder de representar cualquier lenguaje dentro, La maquina de Turing esta definida por una tupla:

$$MT = \{Q, \Sigma, \Gamma, f, q0, b, F\}$$

$Q = \text{ es el conjunto finito de estados}$
$\Sigma = \text{ es el alfabeto de la cadena de entrada}$
$\Gamma = \text{ es el alfabeto de la cinta}$
$q0 \in Q \text{ es el estado inicial}$
$b = \text{ es el espacio en blanco que marca el final de la cadena}$
$F \subset Q \text{ es el conjunto de estados finales}$
$f : Q \times \Gamma \longrightarrow Q \times \Gamma \times \{I, D\}$

Una de las mayores peculiaridades del modelo es la implementación de una cinta, que sustituye la típica entrada de los otros autómatas, esta contiene una cadena finita con los elementos de Σ terminando por un blank “b”, el método de progreso este encargado al cabezal que apuntara al inicio de la cinta cabe resaltar que esta puede tanto leer y escribir caracteres en la cinta, como siempre el modelo comienza en $q0$ donde podremos avanzar entre estados por reglas dadas que siguen el siguiente formato:

$$a \,;\, b \,,\, c$$

$a = \text{es el caracter que se lee de la cinta}$
$b = \text{es el caracter que se escribe}$
$c = \text{define a que direccion se mueve el cavezar como}$
$R = \text{derecha}, L = \text{izquierda y } H = \text{no se mueve}$

En la actualidad las máquinas de Turing se pueden utilizar para simular la lógica de cualquier algoritmo computacional creado explicando el funcionamiento de este, así como también es posible que una Máquina de Turing simule a otra Máquina de Turing llamado “MT Universal”.

Se puede aplicar a muchas áreas de la informática como: Teoría de la Computación, Problema De La Parada, Maquinas Oráculo, Máquinas de Turing como generadoras de Lenguajes, Hipótesis de Church-Turing, etc.

Arellano Granados Angel Mariano
218123444

3. ¿Cómo opera la Maquina de Turing? ¿Qué pasa cuando la MT llega a una solución? ¿Qué pasa cuando el problema no tiene solución?

¿Cómo opera la Maquina de Turing?

Todo depende del cabezal dado a que según el carácter que lea y el estado donde este la maquina definirá que se puede o no hacer, dado a que podemos tener muchos caminos o transiciones en el diagrama, pero solo podremos usar los que contengan el carácter leído por el cabezal.

¿Qué pasa cuando la MT llega a una solución?

Por lo regular es cuando se llega a un estado de aceptación y se coloca un blank al final de la cadena

.

¿Qué pasa cuando el problema no tiene solución?

Este problema puede ser dado por varios errores como que la cinta recibe una cadena que no es compatible con el lenguaje y entra a un estado de no aceptación, que la gramática o el diagrama de transiciones este mal hecho o mal planteado o que se implementa un lenguaje tan complejo que no puede ser expresado con una maquina de Turing lo cual es muy poco probable dado a que es una maquina universal.

4. ¿Puede una Máquina de Turing comportarse como un aceptador de lenguaje? Y en caso de que sea posible, ¿Cómo sería el proceso para aceptar una palabra?

Si las máquinas de Turing pueden reconocer lenguajes y también son capaces de generarlos, una alternativa para el proceso donde una máquina de Turing concluya la aceptación de una cadena es por medio de cambiar todos los caracteres de la cadena recibida por un “Signo de aceptación”, de manera que si al final del proceso logra cambiar todos los caracteres de la cadena original por el signo de aceptación y concluir en un estado de aceptación con un blank al final de la cadena, se considera que la cadena.

Arellano Granados Angel Mariano
218123444

5. Describa cuáles son las principales diferencias entre una Máquina de Turing y un Autómata Finito y un Autómata de Pila.

	Máquina de Turing	Autómata Finito	Autómata de Pila
Diferencias	• Utiliza una cinta como entrada. • Utiliza un cabezal. • Se busca que sea determinista. • Usa gramáticas sin restricciones.	• Puede ser determinista o No determinista. • Usa gramáticas regulares.	• Implementa una pila. • Se busca que sea No determinista • Usa gramáticas sensibles al contexto

6. Escriba un ensayo sobre el "problema de parada" (Halting Problem) en el contexto de Máquinas de Turing.

El problema de la parada o problema de la detención para máquinas de Turing consiste en dada una MT M y una palabra w, determinar si M terminará en un número finito de pasos cuando se ejecuta usando w como entrada.

Alan Turing demostró que este problema es indecidible, ninguna máquina de Turing lo puede resolver.

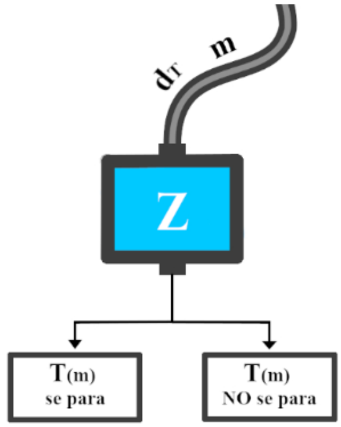
Para demostrar este planteamiento me encontré con varios procesos muy densos en su componente lógico, que me considero incapaz de interpretar uno sin perder datos en el proceso, por lo que expondré el planteamiento que más me gusto completo y concluiré sobre este.

"Para demostrar su imposibilidad usaremos un método bien conocido en el ámbito matemático: la reducción al absurdo. Para argumentar que algo no existe, basta con mostrar como su existencia nos obligaría a afirmar contradicciones absurdas.

Supongamos pues, que tal Máquina de Turing existiese. Llamémosla con el enigmático nombre de **Z**. A esta máquina le introducimos la descripción **dT** de una máquina cualquiera **T** y su cinta **m** como. Suponiendo que esta máquina existe y es capaz de resolver el problema, entonces la máquina debe parar el algún momento dándonos la respuesta a la pregunta: ¿**T(m)** para? Tanto si **T** se para como si no, **Z** debe detenerse devolviéndonos un 1 si se para, y un 0 si no se para.

Arellano Granados Angel Mariano
218123444

$$Z(d_T, m)$$

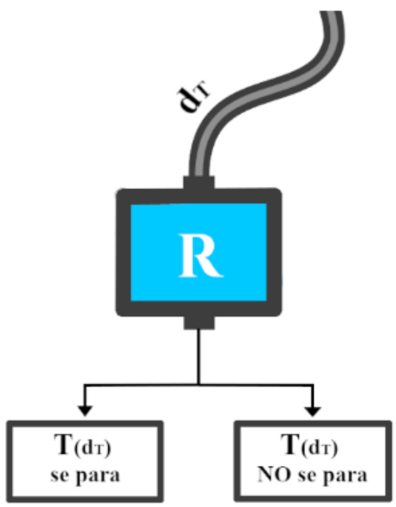


Si nuestra máquina es capaz de esto, será también capaz de responder a la pregunta: ¿**T(dT)** para? Esto es, introducir a la máquina **T** su propia descripción como input en la cinta. Esto puede parecer raro. Nos interesa es si tal cosa sea viable. Y efectivamente, mientras la descripción se encuentre escrita en un lenguaje entendible, una Máquina de Turing no tendría problemas en operar con dicho input. Para nuestra máquina **Z** esto sería así:

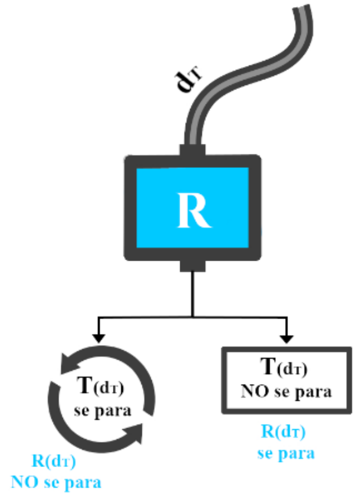
$$Z(d_T, d_T)$$

Vemos que la cinta de **Z** está formada por dos secciones iguales. No nos resultaría difícil suponer que pudiese existir otra Máquina de Turing que operase a la manera en que haría **Z**, pero necesitando solo una de esas secciones **dT**. Tal máquina, que llamaremos **R**, podría hacer un duplicado de **dT** en la cinta.

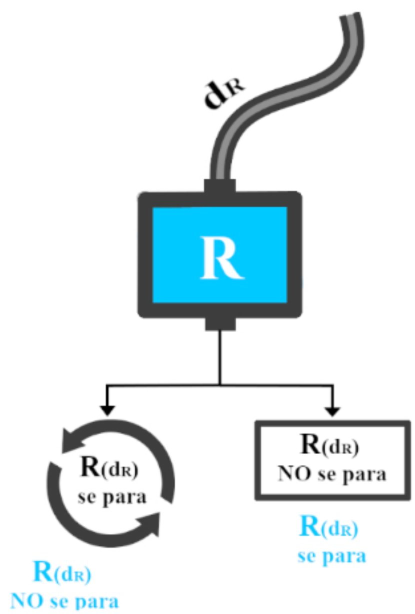
$$Z(d_T, d_T) = R(d_T)$$



R hace el mismo trabajo que **Z**: debe examinar la descripción de una máquina arbitraria **T** y detenerse diciéndonos si tal computación se para 1) o no (0). Podríamos modificar ligeramente a **R** de forma que, antes de detenerse, **R** ejecutara una última operación. Añadimos algún estado más a su configuración de manera que la máquina entre en bucle infinito en caso de que el resultado hubiera sido 1, **T** se para. Ya hemos visto arriba que configurar un bucle infinito es tarea sencilla.



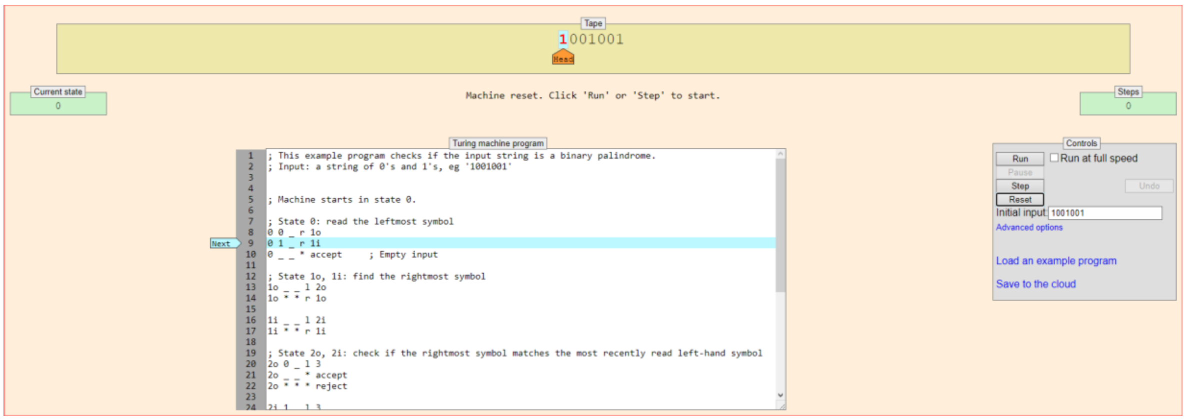
En resumen: si **Z** existiera y resolviese la computabilidad de cualquier máquina **T**, sería viable construir una máquina **R** que se detuviese si **T** no se para, y que entrase en un bucle infinito si **T** se para. Y ahora viene el remate ¿Qué ocurre cuando **R** se enfrenta a su propia descripción? En vez de resolver la computabilidad de introduciendo su descripción **dT**, introduciríamos a **R** la suya propia **dR**. En ese caso acabaríamos en una evidente contradicción absurda: **R se detiene si R no se detiene, y R se detiene cuando R no se detiene.**”



Tras leer varias veces el planteamiento logre entender el problema de la Parada, pues no estaba familiarizado con la idea de usar otras máquinas de Turing como entrada de una máquina de Turing y de los problemas que trae si se inserta la misma máquina sobre si misma.

7. En la siguiente dirección http://morphett.info/turing/turing.html hay un simulador de la máquina de Turing en donde puedes aprender sobre su funcionamiento en tiempo real. Ejecuta un ejemplo para la Máquina de Turing.

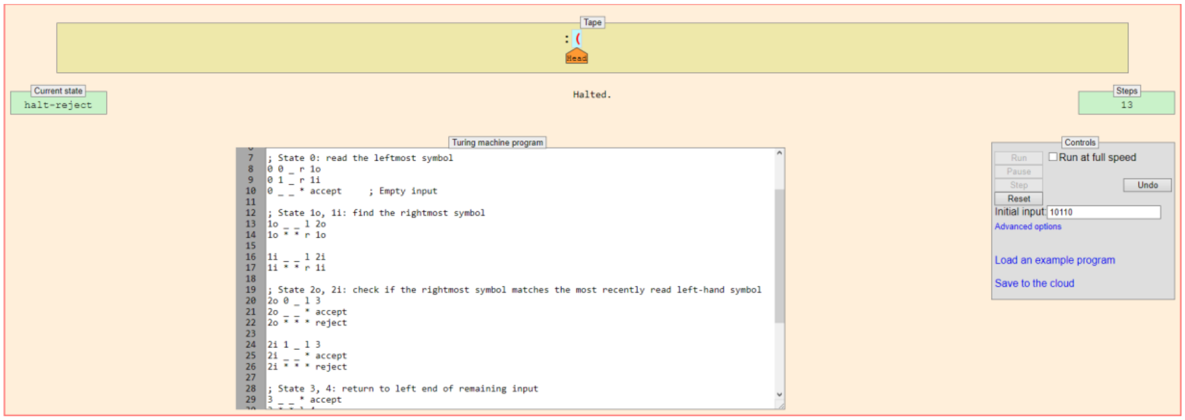
Se ejecutará y analizara el ejemplo de detección de palíndromos con los caracteres ‘0’ y ‘1’, donde nos presenta una interfaz donde podemos visualizar el código, es estado actual, la cinta y la cabeza, así como los controles para cambiar la cadena a analizar y correr el código.



Probaremos el ejemplo predeterminado que corresponde a la cadena ‘1001001’ la cual es un palíndromo, al ejecutar el programa paso a paso podemos ver que la manera en la que funciona es que toma el carácter diferente a blank desde la izquierda lo cambia a blank y se pasa hasta el último carácter diferente a blank desde la derecha lo cambia a blank y si son el mismo se continua en un estado de aceptación, así hasta que ya no quedan caracteres y solo quedan en blank imprime una cara feliz indicando que se acepta la cadena.

Arellano Granados Angel Mariano
218123444

Si ingresamos una cadena que no es un palíndromo como '10110' tras cambiar el primer 1 y ver que el ultimo carácter es un 0 elimina todos los caracteres e imprime una cara triste demostrando que no se acepta la cadena.



9. Escribir un ensayo sobre la paradoja del barbero de Russell y cómo se relaciona ésta con la computabilidad y las Máquinas de Turing.