Teoría de la Computación

Clase 1: Autómatas finitos deterministas (`DFAs`)

Mauro Artigiani

26 julio 2021

Universidad del Rosario, Bogotá

Introducción

Problemas

En Teoría de la Computación, un problema es una función de un conjunto A en un conjunto B.

Decimos que $f\colon A \to \{0,1\}$ es un problema de decisión.

Observación

Sea A un conjunto y $B \subseteq A$. Resolver el problema de decidir si x es un elemento de B es equivalente a construir la función $f_B\colon A \to \{0,1\}$.

Problemas

En Teoría de la Computación, un problema es una función de un conjunto A en un conjunto B.

Decimos que $f\colon A \to \{0, 1\}$ es un problema de decisión.

Observación

Sea A un conjunto y $B \subseteq A$. Resolver el problema de decidir si x es un elemento de B es equivalente a construir la función $f_B\colon A \to \{0, 1\}$.

Un ejemplo de problema de decisión es decidir si una cadena de símbolos es una fórmula. Otro ejemplo es determinar si un polinomio tiene una raíz entera.

Otros tipos de problemas

Sean A y B conjuntos. Una función f de A en B se llama un problema input-output. Por ejemplo, la suma es un problema input output.

Otros tipos de problemas

Sean A y B conjuntos. Una función f de A en B se llama un problema input-output. Por ejemplo, la suma es un problema input output.

Otro ejemplos son los problemas de interacción:
$\text{Input}_0 \to \text{Máquina}_0 \to \text{Output}_0 = \text{Input}_1 \to \text{Máquina}_1 \to \cdots$

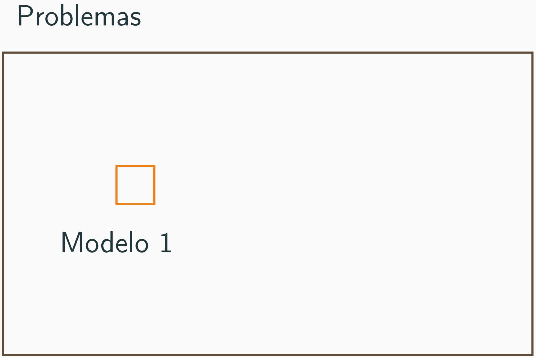
`https://easychair.org/publications/open/DXKk`

Límites de la computación

Cómo demostrar que un problema f no es computable?

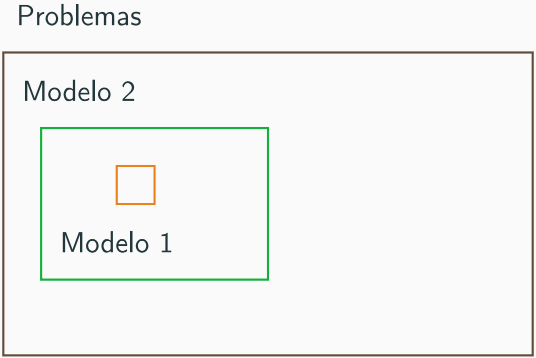
Límites de la computación

Cómo demostrar que un problema f no es computable? Requerimos un modelo de computación.



Límites de la computación

Cómo demostrar que un problema f no es computable? Requerimos un modelo de computación.



Límites de la computación

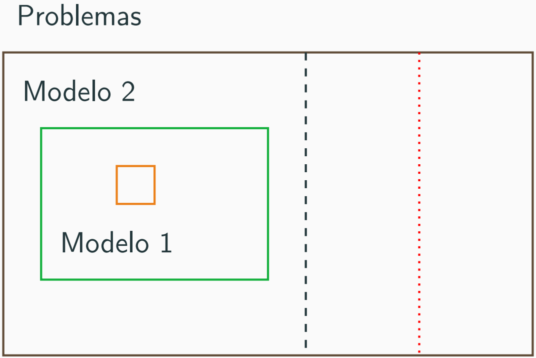
Cómo demostrar que un problema f no es computable? Requerimos un modelo de computación.

Problemas

Límite en principio;

Límites de la computación

Cómo demostrar que un problema f no es computable? Requerimos un modelo de computación.



Límite en principio;

Límite practico (usualmente $T(N) = O(N^{\alpha})$).

Temas del curso

En este curso hablaremos de los siguientes temas:

1. Autómatas finitos deterministas y no deterministas;
2. Gramáticas independientes del contexto y autómatas de pila;
3. Máquinas de Turin e insolubilidad computacional.

Un modelo

Un modelo

En el curso estudiaremos un modelo de un computador sencillo.

Este modelo representa un computador con pocos bits de memoria.

Por ejemplo, el control de una puerta automática, o de un elevador.

Un modelo

En el curso estudiaremos un modelo de un computador sencillo. Este modelo representa un computador con pocos bits de memoria. Por ejemplo, el control de una puerta automática, o de un elevador.

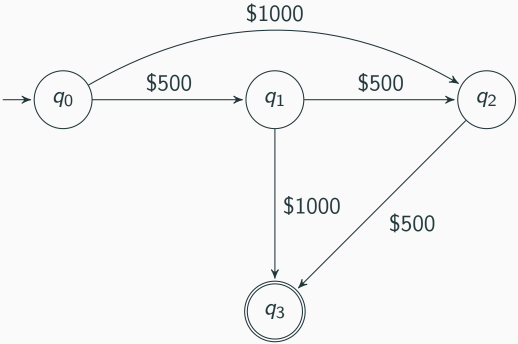
Nuestro modelo tiene una serie de posiciones (o estados), y unas reglas para ir de una posición a otra. Cuando recibe una señal, el computador lee la señal y cambia estado dependiendo de las reglas.

Ejemplo

Una máquina dispensadora de café recibe monedas solo de \$500 y de \$1000. Un café tiene un costo de \$1500. ¿Cómo representar el programa que recibe las monedas y acepta combinaciones de \$1500?

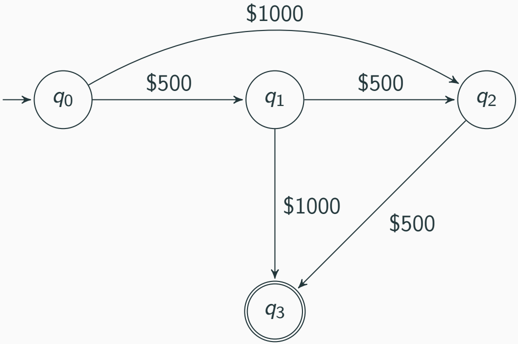
Ejemplo

Una máquina dispensadora de café recibe monedas solo de \$500 y de \$1000. Un café tiene un costo de \$1500. ¿Cómo representar el programa que recibe las monedas y acepta combinaciones de \$1500?



Ejemplo

Una máquina dispensadora de café recibe monedas solo de \$500 y de \$1000. Un café tiene un costo de \$1500. ¿Cómo representar el programa que recibe las monedas y acepta combinaciones de \$1500?



Faltan transiciones: Ejercicio quiz virtual 1.

Tabla de transiciones

Estado	Moneda	Cambia a
q_0	\$500	q_1
q_0	\$1000	q_2
q_1	\$500	q_2
q_1	\$1000	q_3
q_2	\$500	q_3
q_2	\$1000	??
q_3	\$500	??
q_3	\$1000	??

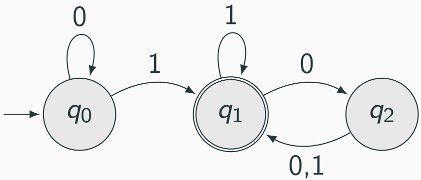
Para una definición de DFA

Ingredientes para la definición

Queremos una definición formal de nuestro modelo sencillo de un computador, que llamaremos un maquina de estados finitos o autómata finito determinista (`DFA`, en inglés).

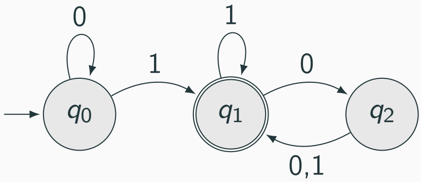
Ingredientes para la definición

Queremos una definición formal de nuestro modelo sencillo de un computador, que llamaremos un maquina de estados finitos o autómata finito determinista (`DFA`, en inglés).



Ingredientes para la definición

Queremos una definición formal de nuestro modelo sencillo de un computador, que llamaremos un maquina de estados finitos o autómata finito determinista (`DFA`, en inglés).



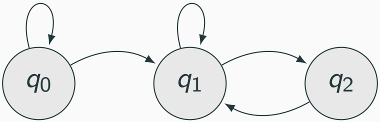
La imagen se llama un diagrama de estados del autómata M. Es un grafo dirigido.

Ingredientes para la definición

Un grafo dirigido G es una pareja $G = (V, E)$, donde $V \neq \emptyset$ es el conjunto de los vértices del grafo y $E \subseteq \{(a, b) \in V \times V\}$ son los arcos o aristas del grafo.

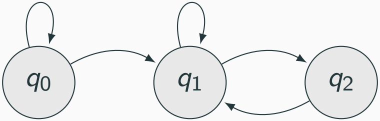
Ingredientes para la definición

Un grafo dirigido G es una pareja $G = (V, E)$, donde $V \neq \emptyset$ es el conjunto de los vértices del grafo y $E \subseteq \{(a, b) \in V \times V\}$ son los arcos o aristas del grafo.



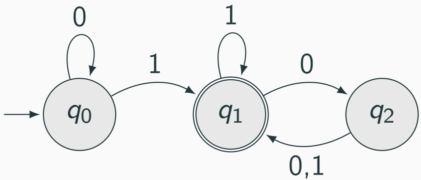
Ingredientes para la definición

Un grafo dirigido G es una pareja $G = (V, E)$, donde $V \neq \emptyset$ es el conjunto de los vértices del grafo y $E \subseteq \{(a, b) \in V \times V\}$ son los arcos o aristas del grafo.



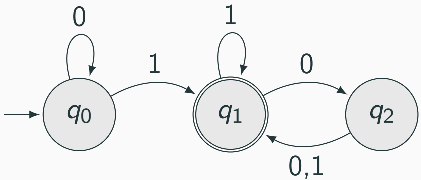
Este grafo corresponde a $V = \{q_0, q_1, q_2\}$ y
$E = \{(q_0, q_0), (q_0, q_1), (q_1, q_1), (q_1, q_2), (q_2, q_1)\}$.

Ingredientes para la definición



El autómata M es más que un grafo.

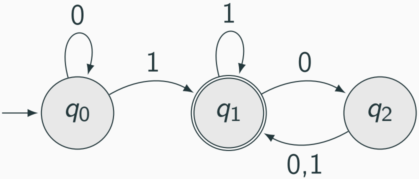
Ingredientes para la definición



El autómata M es más que un grafo. Tiene tres estados, que corresponden a los vértices del grafo, y tienen nombres q_0, q_1 y q_2. El estado q_0 se llama estado inicial. El estado q_1 se llama estado final o de aceptación. A las aristas se las llaman transiciones.

Ingredientes para la definición

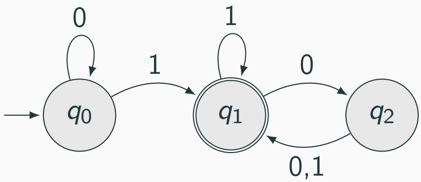
El autómata recibe una cadena de entradas, las lee y al final dice si acepta o rechaza la cadena recibida.



Por ejemplo M acepta la cadena 01, pero rechaza la cadena 00110.

Ingredientes para la definición

El autómata recibe una cadena de entradas, las lee y al final dice si acepta o rechaza la cadena recibida.



Por ejemplo M acepta la cadena 01, pero rechaza la cadena 00110. Cuáles son las cadenas que acepta M?

El lenguaje de un DFA

Dado un alfabeto Σ, las cadenas de elementos en el alfabeto se llaman palabras.

El lenguaje de un DFA

Dado un alfabeto Σ, las cadenas de elementos en el alfabeto se llaman palabras. Una palabra especial es la palabra vacía, que se indica con ε.

El lenguaje de un DFA

Dado un alfabeto Σ, las cadenas de elementos en el alfabeto se llaman palabras. Una palabra especial es la palabra vacía, que se indica con ε.

La longitud de una palabra w es el número de símbolos en Σ que compone w. Por ejemplo, si $\Sigma = \{0, 1\}$ y $w = 101001$, la longitud de w es $|w| = 6$.

El lenguaje de un DFA

Dado un alfabeto Σ, las cadenas de elementos en el alfabeto se llaman palabras. Una palabra especial es la palabra vacía, que se indica con ε.

La longitud de una palabra w es el número de símbolos en Σ que compone w. Por ejemplo, si $\Sigma = \{0, 1\}$ y $w = 101001$, la longitud de w es $|w| = 6$.

Un lenguaje es un conjunto de palabras en un alfabeto.

El lenguaje de un DFA

Dado un alfabeto Σ, las cadenas de elementos en el alfabeto se llaman palabras. Una palabra especial es la palabra vacía, que se indica con ε.

La longitud de una palabra w es el número de símbolos en Σ que compone w. Por ejemplo, si $\Sigma = \{0, 1\}$ y $w = 101001$, la longitud de w es $|w| = 6$.

Un lenguaje es un conjunto de palabras en un alfabeto.

Existe también el lenguaje vació $\emptyset$.

El lenguaje de un DFA

Dado un alfabeto Σ, las cadenas de elementos en el alfabeto se llaman palabras. Una palabra especial es la palabra vacía, que se indica con ε.

La longitud de una palabra w es el número de símbolos en Σ que compone w. Por ejemplo, si $\Sigma = \{0, 1\}$ y $w = 101001$, la longitud de w es $|w| = 6$.

Un lenguaje es un conjunto de palabras en un alfabeto.

Existe también el lenguaje vació $\emptyset$.

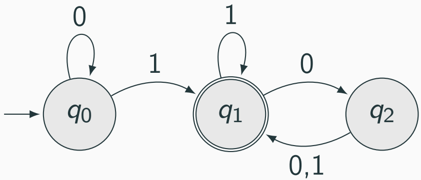
Más sobre lenguaje, cadenas y operaciones entre cadenas en la sección 0.3 del libro de Sipser (y quiz virtual).

El lenguaje de un DFA

Dado un DFA M, llamamos A el conjunto de todas las palabras que el autómata acepta. Decimos que A es el lenguaje de la maquina M, en símbolos $L(M) = A$. Equivalentemente, M reconoce el lenguaje A.

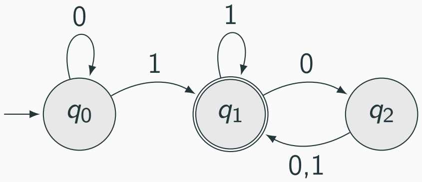
El lenguaje de un DFA

Dado un DFA M, llamamos A el conjunto de todas las palabras que el autómata acepta. Decimos que A es el lenguaje de la maquina M, en símbolos $L(M) = A$. Equivalentemente, M reconoce el lenguaje A.



El lenguaje de un DFA

Dado un DFA M, llamamos A el conjunto de todas las palabras que el autómata acepta. Decimos que A es el lenguaje de la maquina M, en símbolos $L(M) = A$. Equivalentemente, M reconoce el lenguaje A.

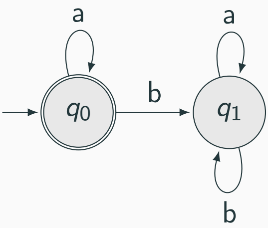


$$L(M) = \{w, w \text{ contiene por lo menos un } 1 \text{ y} \\ \text{hay un número pares de } 0 \text{ después del último } 1\}.$$

Ejemplos

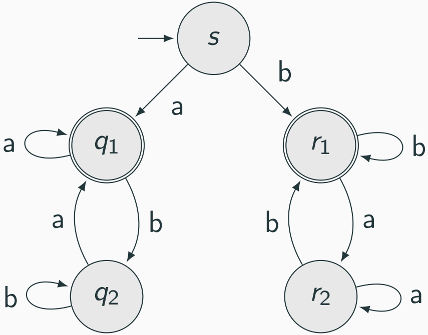
Ejemplos

Consideramos el siguiente diagrama de estado



Cuál es el lenguaje de M?

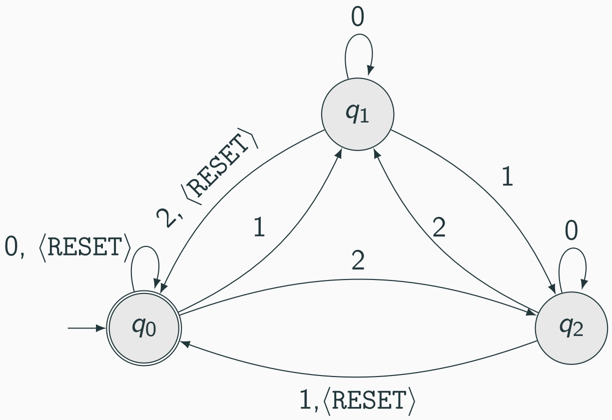
Ejemplos



Cuál es el lenguaje de M?

Ejemplos

Un ejemplo con un alfabeto diferente: $\Sigma = \{\langle\texttt{RESET}\rangle, 0, 1, 2\}$.



Este `DFA` acepta todas palabras que son múltiplos de 3.

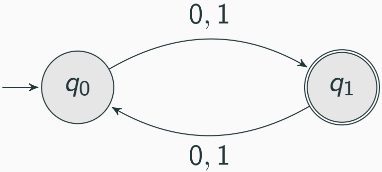
Diseño de DFAs

Ejemplo 1

$L(M) = \{w \in \Sigma^* \colon w \text{ es de longitud impar}\}$

Ejemplo 1

$L(M) = \{w \in \Sigma^*\colon w \text{ es de longitud impar}\}$

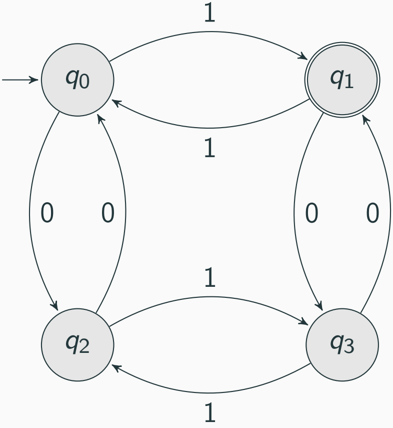


Ejemplo 2

$L(M) = \{w \in \Sigma^* \colon w \text{ tiene un número par de 0s e impar de 1s}\}$

Ejemplo 2

$L(M) = \{w \in \Sigma^* \colon w \text{ tiene un número par de 0s e impar de 1s}\}$

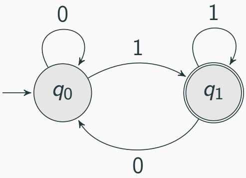


Ejemplos 3 y 4

$L(M) = \{w \in \Sigma^* \colon w \text{ termina en } 1\}$

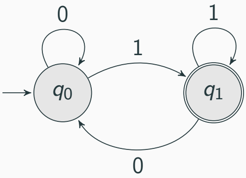
Ejemplos 3 y 4

$L(M) = \{w \in \Sigma^*\colon w \text{ termina en } 1\}$



Ejemplos 3 y 4

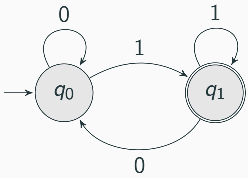
$L(M) = \{w \in \Sigma^*\colon w \text{ termina en } 1\}$



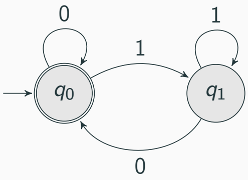
$L(M) = \{w \in \Sigma^*\colon w \text{ es } \varepsilon \text{ o termina en } 0\}$

Ejemplos 3 y 4

$L(M) = \{w \in \Sigma^* \colon w \text{ termina en } 1\}$



$L(M) = \{w \in \Sigma^* \colon w \text{ es } \varepsilon \text{ o termina en } 0\}$

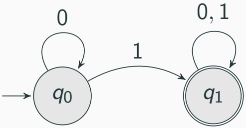


Ejemplos 5 y 6

$L(M) = \{w \in \Sigma^* \colon 1 \text{ es subcadena de } w\}$

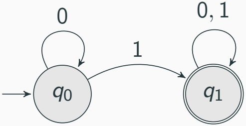
Ejemplos 5 y 6

$L(M) = \{w \in \Sigma^* \colon 1 \text{ es subcadena de } w\}$



Ejemplos 5 y 6

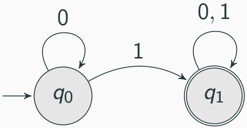
$L(M) = \{w \in \Sigma^* \colon 1 \text{ es subcadena de } w\}$



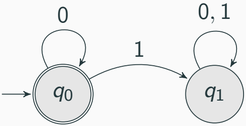
$L(M) = \{w \in \Sigma^* \colon 1 \text{ no es subcadena de } w\}$

Ejemplos 5 y 6

$L(M) = \{w \in \Sigma^* \colon 1 \text{ es subcadena de } w\}$



$L(M) = \{w \in \Sigma^* \colon 1 \text{ no es subcadena de } w\}$



Resumen

Resumen

Hoy aprendimos:

- Qué es un problema en Teoría de la Computación;
- Un modelo de computación y sus límites;
- Como encontrar el lenguaje reconocido por un autómata.
- Como diseñar autómatas que acepten lenguajes dados;