Unidad I: Computabilidad

Reducciones e indecidibilidad

Clase 08 - Lógica para Ciencia de la Computación/Teoría de la Computación

Prof. Miguel Romero

¿Cuáles de los siguientes lenguajes son decidibles?

1. ¿decidir si una MT tiene 324 estados? ✓
2. ¿decidir si una MT toma más de 167 pasos con entrada ε? ✓
3. ¿decidir si una MT toma más de 200 pasos con alguna entrada? ✓
4. ¿decidir si una MT toma más de 934 pasos con todas las entradas? ✓
5. ¿decidir si una MT acepta la entrada 000110? ?
6. ¿decidir si una MT acepta alguna entrada? ?
7. ¿decidir si una MT acepta todas las entradas? ?
8. ¿decidir si una MT acepta una cantidad finita de entradas? ?

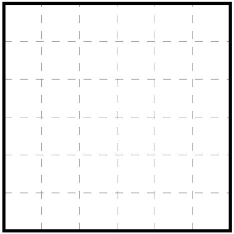
¿Cómo podemos demostrar que los últimos problemas son indecidibles?

Problema A:
Instalar la lámpara del living

Problema B:
Encontrar un maestro

Reducciones: idea

Problema A:
Calculo del área de un cuadrado



Problema B:
Multiplicación de dos números

$\boxed{\text{base} \times \text{altura}}$

Reducción desde **Problema A** a **Problema B**:

Una forma de transformar el **Problema A** al **Problema B**,
de tal manera de que un algoritmo para **B** puede ser usado para resolver **A**.

Un algoritmo para el **Problema A** asumiendo que tenemos una subrutina que resuelve el **Problema B**.

Notación: Decimos que **A** se reduce a **B**.

Reducción desde **Lenguaje** K a **Lenguaje** L:

Un algoritmo para el **Lenguaje** K asumiendo que tenemos una subrutina que resuelve el **Lenguaje** L.

Observaciones:

- Si L es decidible, entonces K es decidible.
- Si K es indecidible, entonces L es indecidible.

Si K se **reduce** a L, la dificultad de K es **menor o igual** a la dificultad de L.

Definamos el siguiente lenguaje sobre $\{0, 1\}$:

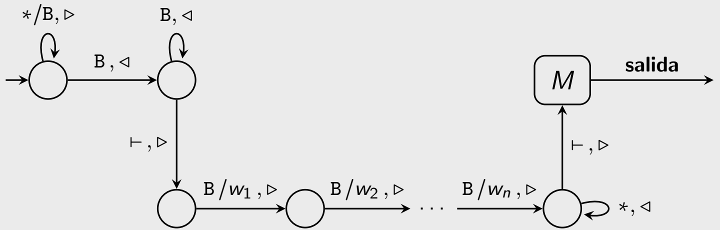
$$\text{Halting}_\varepsilon = \{\langle M \rangle \mid M \text{ es una MT y } M \text{ se detiene con } \varepsilon\}$$

Proposición:

Halting$_\varepsilon$ es indecidible.

Mostraremos una reducción desde Halting a Halting$_\varepsilon$.

Para una MT M y entrada $w = w_1 w_2 \cdots w_n$, definimos la MT $N_{M,w}$ como:



M se detiene con $w \iff N_{M,w}$ se detiene con ε.

Supongamos que tenemos una MT S que **decide** $\text{Halting}_{\varepsilon}$. Describiremos un MT que **decide** Halting, utilizando S como subrutina.

Sobre entrada $z \in \{0,1\}^*$:

1. Verificar que $z = \langle M, w \rangle$ para MT M y $w \in \{0,1\}^*$. Si no, **rechazar**.
2. Simular S sobre $\langle N_{M,w} \rangle$.
3. Si S acepta, **aceptar**.
4. Si S rechaza, **rechazar**.

¿Por qué la reducción es correcta?

Para una palabra fija $x \in \{0,1\}^*$, defina el siguiente lenguaje sobre $\{0,1\}$:

$$\text{Halting}_x = \{\langle M\rangle \mid M \text{ es una MT y } M \text{ se detiene con } x\}$$

La misma reducción anterior se puede aplicar para este caso. (¿por qué?)

Corolario:

Para todo $x \in \{0,1\}^*$, Halting_x es indecidible.

Formalizaremos cierto tipo de reducciones:

- Son un caso simple de la idea general de reducción.
- Tienen buenas propiedades.
- Todos los resultados de indecidibilidad que veremos se pueden obtener usando este tipo de reducciones.
- Son centrales en complejidad computacional.

Definición:

Sean alfabetos finitos Σ_1 y Σ_2. Sea una función $f : \Sigma_1^* \to \Sigma_2^*$ y una MT $M = (Q, \Sigma_1, \Gamma, q_0, q_f, \delta)$. Decimos que M **computa** f si:

1. Para toda entrada $w \in \Sigma_1^*$, la ejecución $\rho = C_0, \ldots, C_m$ de M sobre w es finita. (M se detiene con w.)
2. $C_m = \vdash \; q_f \, f(w)$.

Idea: M computa f, si para toda entrada $w \in \Sigma_1^*$, la MT M se detiene y deja escrito en la cinta la palabra $f(w)$.

Definición:

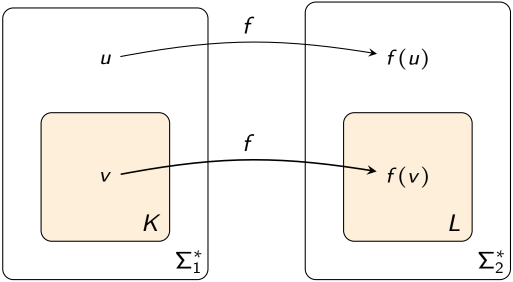
Sean alfabetos finitos Σ_1 y Σ_2, y $f : \Sigma_1^* \to \Sigma_2^*$ una función. Decimos que f es **computable** si existe una MT M que computa f.

Definición:

Sean lenguajes $K \subseteq \Sigma_1^*$ y $L \subseteq \Sigma_2^*$. Una **reducción** (**mucho a uno**) de K a L es una función computable $f : \Sigma_1^* \to \Sigma_2^*$ tal que para toda palabra $w \in \Sigma_1^*$:

$$w \in K \iff f(w) \in L$$

Idea:



Definición:

Sean lenguajes $K \subseteq \Sigma_1^*$ y $L \subseteq \Sigma_2^*$. Una **reducción** (**mucho a uno**) de K a L es una función computable $f : \Sigma_1^* \to \Sigma_2^*$ tal que para toda palabra $w \in \Sigma_1^*$:

$$w \in K \iff f(w) \in L$$

Comentarios:

- La función $f : \Sigma_1^* \to \Sigma_2^*$ no tiene que ser biyectiva.
- Si existe una reducción de K a L, decimos que K se reduce a L y escribimos:

$$K \leq_m L$$

- La relación $\leq_m$ entre lenguajes es **transitiva**:

Si $K \leq_m L$ y $L \leq_m J$, entonces $K \leq_m J$.

(¿por qué?)

Proposición:

Si $K \leq_m L$ y L es decidible, entonces K es decidible.

Demostración:

Sea f la reducción de K a L y M_L la MT que decide L.

La siguiente MT M_K decide K.

Sobre entrada w:

1. Computar $f(w)$.
2. Simular M_L sobre $f(w)$.
3. Si M_L acepta, **aceptar**.
4. Si M_L rechaza, **rechazar**.

Se cumple que:

$$w \in K \iff f(w) \in L \iff M_L \text{ acepta } f(w) \iff M_K \text{ acepta } w$$

Proposición:

Si $K \leq_m L$ y L es decidible, entonces K es decidible.

Corolario:

Si $K \leq_m L$ y K es indecidible, entonces L es indecidible.

Para demostrar que un lenguaje es indecidible,
basta demostrar que otro lenguaje indecidible se reduce a este.

Ojo:

Las propiedades anteriores se cumplen si reemplazamos decidible por recursivamente enumerable. (¿por qué?)

Ya demostramos que $\text{Halting}_\varepsilon$ es indecidible vía una reducción (general) desde Halting.

Demostremos esto mismo usando reducciones mucho a uno.

$$\text{Halting} = \{\langle M, w\rangle \mid M \text{ es una MT}, w \in \{0,1\}^*, M \text{ se detiene con } w\}$$
$$\text{Halting}_\varepsilon = \{\langle M\rangle \mid M \text{ es una MT}, M \text{ se detiene con } \varepsilon\}$$

Reducción:

$$f(\langle M, w\rangle) = \langle N_{M,w}\rangle$$

Notar que:

- f es computable.
- $\langle M, w\rangle \in \text{Halting} \iff \langle N_{M,w}\rangle \in \text{Halting}_\varepsilon$.

Ya demostramos que $\text{Halting}_\varepsilon$ es indecidible vía una reducción (general) desde Halting.

Demostremos esto mismo usando reducciones mucho a uno.

$$\text{Halting} = \{\langle M, w\rangle \mid M \text{ es una MT, } w \in \{0,1\}^*, M \text{ se detiene con } w\}$$
$$\text{Halting}_\varepsilon = \{\langle M\rangle \mid M \text{ es una MT, } M \text{ se detiene con } \varepsilon\}$$

Concluimos que $\text{Halting} \leq_m \text{Halting}_\varepsilon$.

Luego $\text{Halting}_\varepsilon$ es indecidible.

Observación: La misma reducción nos dice que $\text{Halting} \leq_m \text{Halting}_x$ y que Halting_x es indecidible, para cualquier palabra $x \in \{0,1\}^*$.

Recordemos el lenguaje Accept sobre $\{0,1\}$:

$$\text{Accept} = \{\langle M, w\rangle \mid M \text{ es una MT, } w \in \{0,1\}^* \text{ y } M \text{ acepta } w\}$$

Veamos que existe una reducción desde Halting a Accept.

Reducción:

$$f(\langle M, w\rangle) = \langle A_M, w\rangle$$

donde la MT A_M se define como sigue:

Sobre entrada $u \in \{0,1\}^*$:

1. Simular M sobre u.
2. Si M se detiene con u, **aceptar**.

¿Por qué la reducción es correcta?

Recordemos el lenguaje Accept sobre $\{0,1\}$:

$$\text{Accept} = \{\langle M, w\rangle \mid M \text{ es una MT, } w \in \{0,1\}^* \text{ y } M \text{ acepta } w\}$$

Concluimos que Halting $\leq_m$ Accept.

Luego Accept es indecidible.

Definamos el siguiente lenguaje sobre $\{0,1\}$:

$$\text{Non-Empty} = \{\langle M\rangle \mid M \text{ es una MT y } L(M) \neq \varnothing\}$$

Veamos que existe una reducción desde Halting a Non-Empty.

Reducción:

$$f(\langle M, w\rangle) = \langle N'_{M,w}\rangle$$

donde la MT $N'_{M,w}$ se define como sigue:

Sobre entrada $u \in \{0,1\}^*$:

1. Borrar la entrada u.
2. Escribir w en la cinta.
3. Simular M sobre w.
4. Si M se detiene con w, **aceptar**.

¿Por qué la reducción es correcta?

Definamos el siguiente lenguaje sobre $\{0, 1\}$:

$$\text{Non-Empty} = \{\langle M \rangle \mid M \text{ es una MT y } L(M) \neq \varnothing\}$$

Concluimos que Halting $\leq_m$ Non-Empty.

Luego Non-Empty es indecidible.

Definamos el siguiente lenguaje sobre $\{0, 1\}$:

$$\text{All} = \{\langle M \rangle \mid M \text{ es una MT y } L(M) = \{0,1\}^*\}$$

La misma reducción anterior demuestra que Halting $\leq_m$ All. (¿por qué?)

Concluimos que All es indecidible.

Utilizando reducciones mucho a uno

Definamos el siguiente lenguaje sobre $\{0,1\}$:

$$\text{Finite} = \{\langle M\rangle \mid M \text{ es una MT y } L(M) \text{ es finito}\}$$

Veamos que existe una reducción desde Halting a Finite.

Reducción:

$$f(\langle M, w\rangle) = \langle D_{M,w}\rangle$$

donde la MT $D_{M,w}$ se define como sigue:

Sobre entrada $u \in \{0,1\}^*$:

1. Copiar u de cinta 1 a cinta 2.
2. Escribir w en cinta 1.
3. Simular $|u|$ pasos de M sobre w en cinta 1.
4. Si M no se detiene con w en $|u|$ pasos, **aceptar**.
5. Si M se detiene con w dentro de $|u|$ pasos, **rechazar**.

¿Por qué la reducción es correcta?

Utilizando reducciones mucho a uno

Definamos el siguiente lenguaje sobre $\{0, 1\}$:

$$\text{Finite} = \{\langle M \rangle \mid M \text{ es una MT y } L(M) \text{ es finito}\}$$

Concluimos que Halting $\leq_m$ Finite.

Luego Finite es indecidible.