

Teoría de la Computación

Clase 29: Ejemplos de Lenguajes indecidibles y irreconocibles

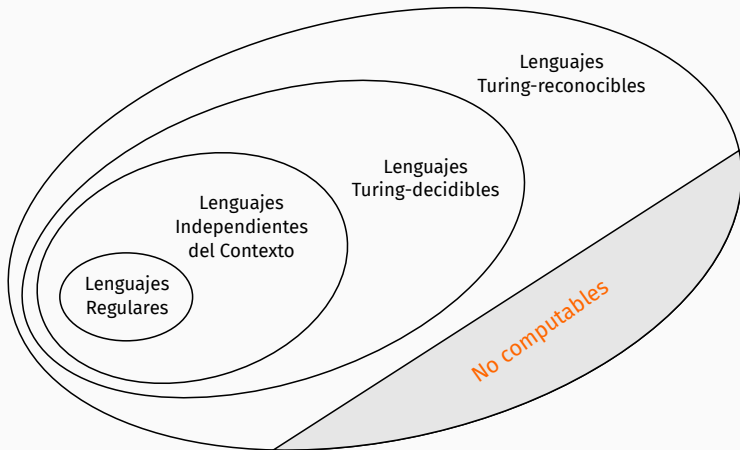
Mauro Artigiani

08 Noviembre 2021

Universidad del Rosario, Bogotá

Introducción

Los lenguajes que hemos visto



Un lenguaje indecidible

Un lenguaje fundamental

Consideramos el lenguaje

$$A_{\text{TM}} = \{ \langle M, w \rangle : M \text{ es una TM tal que } M \text{ acepta } w \}.$$

Un lenguaje fundamental

Consideramos el lenguaje

$$A_{\text{TM}} = \{ \langle M, w \rangle : M \text{ es una TM tal que } M \text{ acepta } w \}.$$

Teorema

El lenguaje A_{TM} no es Turing-decidible.

Un lenguaje fundamental

Consideramos el lenguaje

$$A_{\text{TM}} = \{\langle M, w \rangle : M \text{ es una TM tal que } M \text{ acepta } w\}.$$

Teorema

El lenguaje A_{TM} **no** es Turing-**decidible**.

Nótese que A_{TM} *sí* es Turing-reconocible (¿por qué?)

Demostración: A_{TM} no es Turing-decidible

Supongamos por absurdo que A_{TM} sea Turing-decidible y sea H una máquina de Turing que decida este problema. Es decir:

1. H se detiene en toda entrada;
2. $\langle M, w \rangle \in A_{TM}$ si y solo si H acepta $\langle M, w \rangle$.

Demostración: A_{TM} no es Turing-decidible

Supongamos por absurdo que A_{TM} sea Turing-decidible y sea H una máquina de Turing que decida este problema. Es decir:

1. H se detiene en toda entrada;
2. $\langle M, w \rangle \in A_{TM}$ si y solo si H acepta $\langle M, w \rangle$.

Sea D una TM definida de la siguiente manera:

- 1: **procedure** $D(\langle M \rangle)$
- 2: Correr $H(\langle M, \langle M \rangle \rangle)$
- 3: **if** se obtiene Aceptar **then**
- 4: Rechazar
- 5: **else**
- 6: Aceptar

Demostración: A_{TM} no es Turing-decidible

Para no confundirnos: H por definición acepta $\langle M, w \rangle$ si y solo si M acepta w .

Demostración: A_{TM} no es Turing-decidible

Para no confundirnos: H por definición acepta $\langle M, w \rangle$ si y solo si M acepta w .

En particular podemos elegir $w = \langle M \rangle$. En este caso

D acepta $\langle M \rangle$ sii H no acepta $\langle M \rangle$ sii M no acepta $\langle M \rangle$.

Demostración: A_{TM} no es Turing-decible

Para no confundirnos: H por definición acepta $\langle M, w \rangle$ si y solo si M acepta w .

En particular podemos elegir $w = \langle M \rangle$. En este caso

D acepta $\langle M \rangle$ sii H no acepta $\langle M \rangle$ sii M no acepta $\langle M \rangle$.

Al ser D también una TM, podemos hacer el mismo con su descripción:

D acepta $\langle D \rangle$ sii H no acepta $\langle D \rangle$ sii D no acepta $\langle D \rangle$.

Demostración: A_{TM} no es Turing-decidible

Para no confundirnos: H por definición acepta $\langle M, w \rangle$ si y solo si M acepta w .

En particular podemos elegir $w = \langle M \rangle$. En este caso

D acepta $\langle M \rangle$ sii H no acepta $\langle M \rangle$ sii M no acepta $\langle M \rangle$.

Al ser D también una TM, podemos hacer el mismo con su descripción:

D acepta $\langle D \rangle$ sii H no acepta $\langle D \rangle$ sii D no acepta $\langle D \rangle$.

De esta contradicción sigue que A_{TM} no es Turing-decidible.



Un lenguaje irreconocible

Un criterio para decidibilidad

Lema

Un lenguaje L es Turing-decidible sii L y \bar{L} son Turing-reconocibles.

Un criterio para decidibilidad

Lema

Un lenguaje L es Turing-decidible sii L y \bar{L} son Turing-reconocibles.

\Rightarrow) Supongamos que L sea Turing-decidible. Luego, L es Turing-reconocible.

Un criterio para decidibilidad

Lema

Un lenguaje L es Turing-decidible sii L y \bar{L} son Turing-reconocibles.

\Rightarrow) Supongamos que L sea Turing-decidible. Luego, L es Turing-reconocible. Siendo L decidible existe una máquina de Turing M que lo decide.

Un criterio para decidibilidad

Lema

Un lenguaje L es Turing-decidible sii L y \bar{L} son Turing-reconocibles.

\Rightarrow) Supongamos que L sea Turing-decidible. Luego, L es Turing-reconocible. Siendo L decidible existe una máquina de Turing M que lo decide. Sea N la TM que realiza el siguiente algoritmo:

```
1: procedure  $N(w)$   
2:   Correr  $M(w)$   
3:   if se obtiene Aceptar then  
4:     Rechazar  
5:   else  
6:     Aceptar
```

Un criterio para decidibilidad

Lema

Un lenguaje L es Turing-decidible sii L y \bar{L} son Turing-reconocibles.

\Rightarrow) Supongamos que L sea Turing-decidible. Luego, L es Turing-reconocible. Siendo L decidible existe una máquina de Turing M que lo decide. Sea N la TM que realiza el siguiente algoritmo:

```
1: procedure  $N(w)$   
2:   Correr  $M(w)$   
3:   if se obtiene Aceptar then  
4:     Rechazar  
5:   else  
6:     Aceptar
```

N reconoce a \bar{L} .

Demostración del criterio para decidibilidad

\Leftarrow) Supongamos que ambos L y \bar{L} sean Turing-reconocibles. Sean M y N las TM que los reconocen.

Demostración del criterio para decidibilidad

\Leftarrow) Supongamos que ambos L y \bar{L} sean Turing-reconocibles. Sean M y N las TM que los reconocen. Definimos T que decide a L así:

```
1: procedure  $T(w)$   
2:   Correr en paralelo  $M(w)$  y  $N(w)$   
3:   if  $M$  produce Aceptar then  
4:     Aceptar  
5:   if  $N$  produce Aceptar then  
6:     rechazar
```



Un lenguaje no Turing-reconocible

Podemos utilizar lo que acabamos de demostrar para dar un ejemplo de un lenguaje que no sea Turing-reconocible.

Un lenguaje no Turing-reconocible

Podemos utilizar lo que acabamos de demostrar para dar un ejemplo de un lenguaje que no sea Turing-reconocible.

Corolario

$\overline{A_{TM}}$ no es Turing-reconocible.

Un lenguaje no Turing-reconocible

Podemos utilizar lo que acabamos de demostrar para dar un ejemplo de un lenguaje que no sea Turing-reconocible.

Corolario

$\overline{A_{TM}}$ no es Turing-reconocible.

Demostración

Supongamos por absurdo que $\overline{A_{TM}}$ sea Turing-reconocible.

Un lenguaje no Turing-reconocible

Podemos utilizar lo que acabamos de demostrar para dar un ejemplo de un lenguaje que no sea Turing-reconocible.

Corolario

$\overline{A_{\text{TM}}}$ no es Turing-reconocible.

Demostración

Supongamos por absurdo que $\overline{A_{\text{TM}}}$ sea Turing-reconocible. Hemos dicho antes que A_{TM} es Turing-reconocible.

Un lenguaje no Turing-reconocible

Podemos utilizar lo que acabamos de demostrar para dar un ejemplo de un lenguaje que no sea Turing-reconocible.

Corolario

$\overline{A_{TM}}$ no es Turing-reconocible.

Demostración

Supongamos por absurdo que $\overline{A_{TM}}$ sea Turing-reconocible. Hemos dicho antes que A_{TM} es Turing-reconocible. Entonces, por el lema anterior sigue que A_{TM} es Turing-decidible, lo que es absurdo.

De la contradicción sigue que $\overline{A_{TM}}$ no es Turing-reconocible. \square

Si \bar{L} es Turing-reconocible, decimos que L es **co-Turing-reconocible**. El corolario también se puede expresar como A_{TM} no es co-Turing-reconocible.

Resumen

Hoy aprendimos:

- Un ejemplo de lenguaje no Turing-decidible;
- Un ejemplo de lenguaje no Turing-reconocible;
- La relación entre lenguajes Turing-decidibles, Turing-reconocibles y co-Turing-reconocibles.