



Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencias de la Computación
Matías Fernández - matias.fernandez@uc.cl

IIC2213 - Lógica para ciencia de la computación

Ayudantía 4 - Viernes 14 de Abril del 2023

Problema 1. Si L_1, L_2 son decidibles demuestre que:

- a) $L_1 \cap L_2$ es decidible
- b) $L_1 \cup L_2$ es decidible
- c) $\overline{L_1}$ es decidible

Solución:

Sean M_1 y M_2 las máquinas de turing tales que $L(M_1) = L_1$ y $L(M_2) = L_2$.

- a) Podemos hacer una máquina que primero simule M_1 , si rechaza entonces rechazamos, si acepta entonces pasamos a simular M_2 con el input original. Si esta última rechaza entonces rechazamos, si acepta entonces aceptamos globalmente.
- b) Similar a la parte anterior, podemos simular la máquina M_1 , si acepta entonces aceptamos globalmente, si rechaza entonces pasamos a simular la máquina M_2 . Si esta última rechaza entonces rechazamos, si acepta entonces aceptamos globalmente.
- c) Podemos simular la máquina M_1 y si acepta, entonces rechazamos globalmente, si rechaza entonces aceptamos globalmente.

Problema 2. Sean L_1, L_2 lenguajes tales que L_1 es reducible a L_2 . Pruebe que

- a) Si L_2 es decidible entonces L_1 es decidible
- b) Si L_1 es indecidible entonces L_2 es indecidible

Solución:

Vamos a solo demostrar a), ya que es equivalente a b).

Como existe f que lleva de L_1 a L_2 que es computable por una máquina de turing. Llamemos a esta M_f . Y como L_2 es decidible, llamaremos a la máquina que lo decide M_2 . Ahora podemos construir una máquina M_1 tal que al input que se le entregue, lo pase por M_f , como sabemos que M_f computa f entonces nos entregará una palabra en L_2 si y sólo si w (input) pertenecía a L_1 . Ahora, podemos simular M_2 con input $f(w)$.

Problema 3. Demuestre que los siguientes lenguajes son indecidibles

- a) $DD = \{(\mathcal{M}_1, \mathcal{M}_2) \mid \mathcal{M}_1 \text{ se detiene con entrada } \text{cod}(\mathcal{M}_1) \text{ y } \mathcal{M}_2 \text{ se detiene con entrada } \text{cod}(\mathcal{M}_2)\}$
- b) $A = \{(\text{cod}(\mathcal{M}), w) \mid \mathcal{M} \text{ es una MT determinista tal que } \mathcal{M} \text{ acepta } w\}$

c) $L_1 = \{\text{cod}(\mathcal{M}) \mid \mathcal{M} \text{ es una MT determinista tal que } L(\mathcal{M}) = \emptyset\}$

Solución:

a) Definimos la función computable $f(w) = (w, w)$. O sea, repetir w . Vamos a demostrar que esta función es una reducción del lenguaje visto en clases DG (diagonal) a DD . Queremos demostrar que $w \in DG \Leftrightarrow f(w) \in DD$.

\Rightarrow Si $w \in DG$ entonces $w = \text{cod}(\mathcal{M})$ tal que \mathcal{M} se detiene con $\text{cod}(\mathcal{M})$. Luego es claro que $f(w) = (w, w) \in DD$.

\Leftarrow Si $f(w) \in DD$ entonces $f(w) = (w, w)$ es tal que $w = \text{cod}(\mathcal{M})$ tal que \mathcal{M} se detiene con $\text{cod}(\mathcal{M})$ y $w = \text{cod}(\mathcal{M})$ tal que \mathcal{M} se detiene con $\text{cod}(\mathcal{M})$. Pero esto es redundante, y se tiene por definición de DG que $w \in DG$.

Finalmente, $DG \leq_T DD$ pero DG es indecidible por lo tanto, DD también lo es.

b) Probaremos que $H \leq_T A$. Definamos f tal que $f(w') = f((\text{cod}(\mathcal{M}), w)) = (\text{cod}(\mathcal{M}'), w)$ donde \mathcal{M}' es una máquina igual a \mathcal{M} salvo que todos sus estados son finales. Es claro que se puede hacer esto con una máquina de turing, ahora basta demostrar que es una reducción de H a A .

\Rightarrow Si $w' \in H$ entonces $w' = (\text{cod}(\mathcal{M}), w)$ donde la máquina \mathcal{M} se detiene con input w . Luego es claro que si modificamos \mathcal{M} a \mathcal{M}' para que todos sus estados sean finales entonces \mathcal{M}' con input w se debe detener en algún estado, y por ende acepta. Así $f(w')$ pertenece a A .

\Leftarrow Si $f(w') = f((\text{cod}(\mathcal{M}), w)) = (\text{cod}(\mathcal{M}'), w)$ pertenece a A . Por lo tanto \mathcal{M}' acepta a w . Ahora bien, si acepta a w y \mathcal{M}' es igual a \mathcal{M} salvo estados finales. Por lo tanto \mathcal{M} debe al menos parar con w . Entonces w' pertenece a H .

Finalmente, como H es indecidible entonces A es indecidible.

c) Vamos a asumir que L_1 es decidable y vamos a construir una máquina de turing que decide a A .

Primero verificamos que el input sea como corresponde $w' = (\text{cod}(\mathcal{M}), w)$. Luego modificamos el input para que pase a ser la máquina \mathcal{M}' la cual borra cualquier input que le den e inserta w en la cinta. Y finalmente le pasamos esta $\text{cod}(\mathcal{M}')$ a la máquina que decide a A , M_A . Si acepta, rechazamos y si rechaza, aceptamos. Notemos que si M_A rechaza a $\text{cod}(\mathcal{M}')$ solo puede darse debido a que la máquina \mathcal{M} acepte a w , y viceversa. Pero esto quiere decir que A es decidable por la máquina que acabamos de crear, contradicción.