Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

IIC2213 — Lógica para Ciencia de la Computación — 2025-01

TAREA 3

Pregunta 1

$D = \{\langle M_1, w_1, M_2, w_2\rangle \mid M_1, M_2 \text{ son MTs}, w_1, w_2 \in \{0,1\}^*, M_1 \text{ se detiene con } w_1, M_2 \text{ no se detiene con } w_2\}$

(a) (1.0 pts) Sean lenguajes K y L. Demuestre que si $K \leq_m L$, entonces $\overline{K} \leq_m \overline{L}$.

(b) (2.5 pts) Demuestre que D no es recursivamente enumerable.

(c) (2.5 pts) Demuestre que $\overline{D}$ no es recursivamente enumerable.

(Hint: Recuerde que $\overline{Halting}$ no es recursivamente enumerable.)

Solución:

(a) Sea $K \leq_m L$, esto quiere decir que existe una función computable $f : \Sigma_1^* \to \Sigma_2^*$ tal que para toda palabra $w \in K \iff f(w) \in L$. (K y L están sobre los alfabetos Σ_1 y Σ_2, respectivamente.)

Buscamos una nueva función computable g tal que se cumpla que $w \in \overline{K} \iff g(w) \in \overline{L}$.

Resulta que podemos usar como función g la misma función f, ya que se cumple:

$$w \in \overline{K} \iff w \notin K \iff f(w) \notin L \iff f(w) \in \overline{L}$$

Como f es computable, hemos encontrado un reducción mucho a uno desde $\overline{K}$ a $\overline{L}$. Por lo tanto, $\overline{K} \leq_m \overline{L}$.

(b) Para demostrar que D no es recursivamente enumerable debemos reducirlo desde otro lenguaje que no sea recursivamente enumerable. Demostraremos así que $\overline{\text{Halting}} \leq_m D$.

Recordemos que

$\overline{\text{Halting}} = \{\langle M, w\rangle \mid M \text{ MT}, w \in \{0,1\}^*, M \text{ no se detiene con } w\} \cup \{z \mid z \text{ no tiene le formato } \langle M, w\rangle\}$.

Para realizar la reducción a D desde $\overline{\text{Halting}}$ consideramos la función computable $f : \{0,1\}^* \to \{0,1\}^*$ tal que:

- Si z no está en el formato $\langle M, w\rangle$, entonces $f(z) = x \in D$, donde $x \in \{0,1\}^*$ es cualquier palabra fija que pertenece a D.
- Si $z = \langle M, w\rangle$, entonces definimos $f(\langle M, w\rangle) = \langle M_{stop}, \varepsilon, M, w\rangle$, donde M_{stop} es una MT fija que frente a toda entrada se detiene y acepta.

Veamos que $z \in \overline{\text{Halting}}$ si y sólo si $f(z) \in D$. Supongamos que $z \in \overline{\text{Halting}}$. Tenemos dos casos. Si z no tiene el formato $\langle M, w\rangle$, por construcción, tenemos que $f(z) \in D$. Si $z = \langle M, w\rangle$ para una MT M y palabra $w \in \{0,1\}^*$, sabemos que M no se detiene con w. Tenemos que $f(z) = f(\langle M, w\rangle) = \langle M_{stop}, \varepsilon, M, w\rangle$. Como M_{stop} se detiene con ε y M no se detiene con w, concluimos que $f(z) \in D$.

Supongamos que $z \notin \overline{\text{Halting}}$. Esto implica que $z = \langle M, w\rangle$ para una MT M y palabra $w \in \{0,1\}^*$, tal que M se detiene con w. Por definición de f, tenemos que $f(z) = f(\langle M, w\rangle) = \langle M_{stop}, \varepsilon, M, w\rangle$. Como M se detiene con w, no se satisface la condición en la definición del lenguaje D, y entonces $f(z) \notin D$.

Observación: También se considera bueno ignorar los casos de borde cuando z no tiene el formato correcto. Es decir, definir $\overline{\text{Halting}}$ como $\overline{\text{Halting}} = \{\langle M, w\rangle \mid M \text{ MT}, w \in \{0,1\}^*, M \text{ no se detiene con } w\}$, y trabajar con esa definición.

(c) Para demostrar que $\overline{D}$ no es recursivamente enumerable debemos demostrar que $\overline{\text{Halting}} \leq_m \overline{D}$.

Una primera solución es demostrar que Halting $\leq_m D$ y luego aplicar el ítem a) y así concluir $\overline{\text{Halting}} \leq_m \overline{D}$.

Recordemos que

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

Para realizar la reducción a D desde Halting consideramos la función computable $f : \{0,1\}^* \to \{0,1\}^*$ tal que:

- Si z no está en el formato $\langle M, w\rangle$ entonces $f(z) = \varepsilon \notin D$. (acá basta poner cualquier palabra que no esté en D, por ejemplo, ε.)
- Si $z = \langle M, w\rangle$, entonces $f(\langle M, w\rangle) = \langle M, w, M_{loop}, \varepsilon\rangle$, donde M_{loop} es un MT fija que frente toda entrada no se detiene. (siempre se puede definir tal MT.)

Veamos que $z \in$ Halting si y sólo si $f(z) \in D$. Supongamos que $z \in$ Halting. Entonces $z = \langle M, w\rangle$ para una MT M y palabra $w \in \{0,1\}^*$, tal que M se detiene con w. Por definición de f, tenemos que $f(z) = f(\langle M, w\rangle) = \langle M, w, M_{loop}, \varepsilon\rangle$. Como M se detiene con w y M_{loop} no se detiene con ε, obtenemos que $f(z) \in D$.

Supongamos que $z \notin$ Halting. Tenemos dos casos. Si z no está en el formato $\langle M, w\rangle$, por construcción, tenemos que $f(z) \notin D$. Si $z = \langle M, w\rangle$ para una MT M y palabra $w \in \{0,1\}^*$, sabemos que M no se detiene con w. Tenemos que $f(z) = f(\langle M, w\rangle) = \langle M, w, M_{loop}, \varepsilon\rangle$. Como M no se detiene con w, no se satisface la condición en la definición del lenguaje D, y entonces $f(z) \notin D$.

Una segunda solución es no ocupar el ítem (a) y construir directamente una reducción mucho a uno de $\overline{\text{Halting}}$ a $\overline{D}$. La misma reducción que hicimos anteriormente nos sirve. Notar que la demostración del item (a) nos dice que efectivamente la misma reducción debería servir.

Observación: Se considera bueno ignorar el caso cuando z no tiene el formato correcto (como se hizo en clases al reducir desde Halting).

Distribución de puntaje:

a)
- 1.0 pts si menciona la función computable y demuestra $\overline{K} \leq_m \overline{L}$

b)
- 0.5 pts por decir el lenguaje desde el cual se hace la reducción y mencionar que se necesita una función computable para realizarla.
- 1.5 pts por la construcción de la función para la reducción.
- 0.5 pts por demostrar correctitud o explicar por qué por construcción la reducción es correcta.

c)
- 0.5 pts por decir el lenguaje desde el cual se hace la reducción y mencionar que se necesita una función computable para realizarla.
- 1.5 pts por la construcción de la función para la reducción.
- 0.5 pts por finalizar la demostración y demostrar correctitud o explicar por qué por construcción la reducción es correcta.

Pregunta 2

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

$$\text{Write} = \{\langle M \rangle \mid M \text{ es una MT, en la ejecución de } M \text{ con } \varepsilon, \text{ en algún momento se escribe 0 en la posición 1}\}.$$

Demuestre que Write es indecidible. En su demostración **debe** utilizar reducciones mucho a uno: demostrar que $K \leq_m \text{Write}$, donde K es algún lenguaje indecidible visto en clases.

Solución:

Se demostrará que Halting se puede reducir a Write, es decir, Halting $\leq_m$ Write. Recordar que:

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

Definimos la reducción f como:

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

donde $W_{M,w}$ es una MT que hace lo siguiente:

Definición de $W_{M,w}$:

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

(1) Borrar u de la cinta.

(2) Siendo $w = w_1 \cdots w_n$, escribir $\#w_1 ... w_n$ entre las posiciones 2 y $n+2$ de la cinta, donde $\#$ es un nuevo símbolo. La cinta quedaría visualmente de la siguiente manera:

$\vdash$	B	$\#$	w_1	...	w_n
0	1	2	3	...	$n+2$

(3) Poner la cabeza de lectura en la posición 3.

(4) Simular M con w, reemplazando en las transiciones $\vdash$ por $\#$.

(5) Si M se detiene con w, escribir un 0 en la posición 1 y aceptar.

Veamos que la reducción es correcta, es decir, que se cumple

$$\langle M, w \rangle \in \text{Halting} \iff \langle W_{M,w} \rangle \in \text{Write}.$$

Supongamos que $\langle M, w \rangle \in$ Halting. Tenemos que M se detiene con w. De esta manera, cuando ejecutamos $W_{M,w}$ con entrada igual a ε, el paso (4) termina y alcanzamos el paso (5). En particular, se escribe un 0 en la posición 1. Concluimos que $\langle W_{M,w} \rangle \in$ Write.

Supongamos ahora que $\langle M, w \rangle \notin$ Halting. Tenemos que M no se detiene con w. De esta manera, al ejecutar $W_{M,w}$ con ε, la MT se queda en loop infinito en el paso (4) y jamás llega al paso (5). Recordar que siempre asumimos que un MT nunca se mueve a la izquierda del símbolo inicio. Esto implica que al simular M con w, nunca nos movemos a la izquierda del símbolo $\#$. Luego en el paso (4) nunca se lee la posición 1, en particular, nunca se escribe un 0 en la posición 1. Obtenemos que $\langle W_{M,w} \rangle \notin$ Write.

Distribución de Puntaje:

- 4 puntos por reducción correcta
- 2 puntos por demostrar la correctitud