Modelos Avanzados de Computación

Relación de Ejercicios 2

9 (Manual)

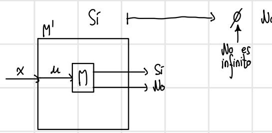
PALÍNDROMO (M)

- •) No es decidible porque es una propiedad no trivial del lenguaje (Teorema de Rice).
- •) Sí es semidecidible porque elegimos de forma no determinista un palíndromo y simulamos la MT con una MT Universal.

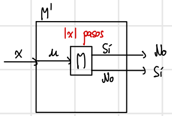
12 c) INFINITO (M')

- •) No es decidible porque es una propiedad no trivial del lenguaje (Teorema de Rice).
- •) Los semidecidibles suelen ser de la forma "existe...". Este no es semidecidible.

C-UNIVERSAL $(M, u) \propto$ INFINITO (M')



Esta reducción no nos sirve



1. Sobre el alfabeto $A = \{a, b, c\}$, calcular las palabras $C(143)$ y $C(100)$. Calcular los enteros $N(aabc)$ y $N(bac)$.

$N(\varepsilon) = 0$; $N(a) = 1$; $N(b) = 2$; $N(c) = 3$

•) $C(143)$

$143 = 3 \cdot 47 + 2$; $47 = 3 \cdot 15 + 2$; $15 = 3 \cdot 4 + 3$; $4 = 3 \cdot 1 + 1$; $1 = 3 \cdot 0 + 1$

$11322 \longrightarrow aacbb = C(143)$

•) $C(100)$

$100 = 3 \cdot 33 + 1$; $33 = 3 \cdot 10 + 3$; $10 = 3 \cdot 3 + 1$; $3 = 3 \cdot 0 + 3$

$0101 \longrightarrow caca = C(100)$

•) $N(aabc) = 3 \cdot 3^0 + 2 \cdot 3 + 1 \cdot 3^2 + 1 \cdot 3^3 = 3 + 6 + 9 + 27 = 45$

•) $N(bac) = 3 \cdot 3^0 + 1 \cdot 3 + 2 \cdot 3^2 = 3 + 3 + 18 = 24$

3. Construir una máquina de Turing que, dada una entrada w, la convierte en la salida $w111w$.

MT con 2 cintas $a \in A$, siendo A el alfabeto de entrada

$\delta(q_0, a, \#) = (q_0, a, a, D, D)$
$\delta(q_0, \#, \#) = (q_1, \#, 1, S, D)$
} Copiamos la palabra de la 1ª cinta en la 2ª cinta y escribimos un 1 al final en la 2ª cinta

$\delta(q_1, \#, \#) = (q_1, \#, 1, I, D)$
$\delta(q_1, a, \#) = (q_2, a, 1, I, D)$
} Escribimos dos 1's más al final de la 2ª cinta

$\delta(q_2, a, \#) = (q_2, a, \#, I, S)$
$\delta(q_2, \#, \#) = (q_3, \#, \#, D, S)$
} Movemos el cabezal de la primera cinta al principio

$\delta(q_3, a, \#) = (q_3, a, a, D, D)$
$\delta(q_3, \#, \#) = (q_f, \#, \#, S, S)$
} Copiamos de nuevo la palabra de la 1ª cinta en la 2ª cinta, terminamos y aceptamos

5. Sean $L_1, \ldots, L_k$ $(k \geq 2)$ un conjunto de lenguajes sobre el alfabeto A tales que:

(a) Para cada $i \neq j$, tenemos que $L_i \cap L_j = \emptyset$.

(b) $\bigcup_{i=1}^{k} L_i = A^*$.

(c) $\forall i \in \{1, \ldots, k\}$, el lenguaje L_i es r.e.

Demostrar que $\forall i \in \{1, \ldots, k\}$, el lenguaje L_i es recursivo.

Tomamos L_i, i fijo. Por hipótesis, L_i es r.e. y $\overline{L_i} = \bigcup_{j \neq i}^{k} L_j$. Veamos que la unión finita de lenguajes r.e. es r.e.: Sea M_j la MT que acepta L_j, $\forall j \neq i$. Construimos una MT con $k-1$ cintas, M, donde cada cinta funciona como M_j. Dada una entrada w, si alguna M_j, con $j \neq i$, acepta w, entonces M acepta w. En caso contrario (Ninguna M_j acepta), M rechaza w. De esta forma, M acepta todos los casos positivos, luego, la unión finita de lenguajes r.e. es r.e. En conclusión: $\left\{ \begin{array}{l} L_i \text{ es r.e.} \\ \overline{L_i} = \bigcup_{j \neq i} L_j \text{ es r.e.} \end{array} \right\} \underset{\text{Teoría}}{\Rightarrow} L_i$ es recursivo. Como i es arbitrario, L_i es recursivo, $\forall i \in \{1, \ldots, k\}$. ∎

6. Sea L un lenguaje recursivamente enumerable, pero no recursivo. Considérese el lenguaje

$$L' = \{0w \mid w \in L\} \cup \{1w \mid w \notin L\}$$

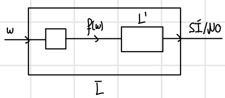
¿Puede asegurarse que L' o su complementario son recursivos, recursivamente enumerables o no recursivamente enumerables?

- $\overline{L}$ no puede ser r.e. ya que, si lo fuera, entonces L sería recursivo, lo cual no es cierto por hipótesis. Así, $\overline{L}$ no es r.e.

- Veamos que L' no es r.e. reduciendo $\overline{L}$ a L'. Dada una palabra $w \in A^*$, $f(w) = 1w$. f es una función totalmente calculable ya que existe una MT que la calcula y siempre para.

  *) Si $w \in \overline{L}$, es decir, $w \notin L \Rightarrow f(w) = 1w \in L'$

  *) Si $w \notin \overline{L}$, es decir, $w \in L \Rightarrow f(w) = 1w \notin L'$

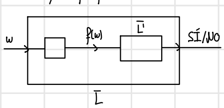


Por tanto, $\overline{L}$ se reduce a L'. Como $\overline{L}$ no es r.e., entonces L' no es r.e.

- $\overline{L'} = \{0w \mid w \notin L\} \cup \{1w \mid w \in L\} \cup \{$Palabras que empiezan por caracteres diferentes a 0 o 1$\}$. Veamos que $\overline{L'}$ no es r.e. reduciendo $\overline{L}$ a $\overline{L'}$. Dada una palabra $w \in A^*$, $f(w) = 0w$. f es una función totalmente calculable ya que existe una MT que la calcula y siempre para.

  *) Si $w \in \overline{L}$, es decir, $w \notin L \Rightarrow f(w) = 0w \in \overline{L'}$

  *) Si $w \notin \overline{L}$, es decir, $w \in L \Rightarrow f(w) = 0w \notin \overline{L'}$



Por tanto, $\overline{L}$ se reduce a $\overline{L'}$. Como $\overline{L}$ no es r.e., entonces $\overline{L'}$ no es r.e.

7. Estudiar si las clases de lenguajes recursivos y recursivamente enumerables son cerradas para las siguientes operaciones:

(a) Unión.

(b) Intersección.

(c) Concatenación.

(d) Clausura.

(e) Homomorfismo.

(f) Homomorfismo inverso.

Nota: Una clase de lenguajes $\mathcal{A}$ es *cerrada* para una operación si siempre que se lleve a cabo la operación con un lenguaje de $\mathcal{A}$ el resultado también pertenece a dicha clase, $\mathcal{A}$

(a) Unión.

Recursivos:

- La unión finita de lenguajes recursivos es recursiva. Basta con construir una MT que, dada una entrada w, simule el comportamiento de todas las MT que aceptan los lenguajes de la familia, y acepta la entrada si una de dichas MT la acepta y rechaza cuando todas la rechazan.
- La unión numerable de lenguajes recursivos no es recursiva. Si construimos una MT para ello de la misma forma anterior, dicha MT tendría que simular el comportamiento de infinitas MT y en los casos negativo, la MT no podría rechazar la entrada hasta que las infinitas MT la rechacen.

Recursivamente enumerables: La unión numerable de lenguajes r.e. es r.e. Construimos una MT que acepte la entrada w si una de las MT que acepta un lenguaje de la familia acepta w. Primero ejecuta la primera MT 1 paso, luego las dos primeras MT 2 pasos, etc. Así de forma secuencial hasta que alguna de las MT acepte la entrada. En caso contrario, la MT puede rechazar o ciclar.

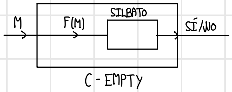
(b) Intersección.

	Recursivo	Recursivamente Enumerable
(a) Unión	Finita	Numerable
(b) Intersección	Finita	Finita
(c) Concatenación	Sí	Sí
(d) Clausura	Sí	Sí
(e) Homomorfismo	Sí	Sí
(f) Homomorfismo inverso	Sí	Sí

10. Supongamos que tenemos MT con dos tipos de estados: *campana* y *silbato*. Determinar que saber si una MT entrará alguna vez en un estado *silbato* es indecidible. ¿Es semidecidible?

Reducimos el problema C-EMPTY al problema de ver si una MT entrará en estado silbato (SILBATO). Dada como entrada una máquina de Turing M, construimos una máquina de Turing F(M) en la que cambiamos todos los estados finales de M por estados silbato y todos los no finales de M por estados campana.

- Si M acepta un lenguaje no vacío $\Rightarrow$ F(M) entra en estado silbato
- Si M no acepta un lenguaje no vacío $\Rightarrow$ F(M) entra en estado campana



Así, hemos reducido C-EMPTY a nuestro problema. Como C-EMPTY es indecidible, entonces SILBATO es indecidible. Por otro lado, SILBATO es semidecidible

11. (MANUAL) Demostrar que es indecidible (no recursivo) saber si una MT termina escribiendo un 1 cuando comienza con una cinta completamente en blanco.

Mirar en el MANUAL.

12. Determinar si los siguientes problemas son decidibles, semidecidibles o no semidecidibles:

(a) Determinar si el lenguaje de una MT contiene, al menos, dos palabras distintas.

(b) Determinar si el lenguaje reconocido por una MT es finito.

(c) (MANUAL) Determinar si el lenguaje reconocido por una MT es infinito.

(d) Determinar si el lenguaje de una MT es independiente del contexto.

17. Sea el problema de determinar si una máquina de Turing acepta a lo más 100 palabras. Determinar si es decidible, semidecidible o no semidecidible. Justificar la respuesta.

- Por el Teorema de Rice, el problema no es decidible.
- Pensamos en el complementario (aceptar más de 100, al menos 101), el cual es semidecidible. Por lo tanto, este problema no es semidecidible (si lo fuera, sería recursivo pero hemos visto que no lo es).

16. Supongamos que tenemos 4 lenguajes L_1, L_2, L_3, L_4 de tal manera que existe una reducción de L_1 a L_2, de L_2 a L_3 y de L_4 a L_3. Para cada una de las siguientes afirmaciones, indicar si para todos los casos son CIERTAS, para todos los casos son FALSAS o si son POSIBLES (en algunos casos son ciertas y, en otros, son falsas).

Todo lo bueno que se pueda decir de la derecha, se puede decir de la izquierda
Todo lo malo que se pueda decir de la izquierda, se puede decir de la derecha

(a) L_1 es recursivamente enumerable pero no recursivo, y L_3 es recursivo Falso

(b) L_1 no es recursivo y L_4 no es recursivamente enumerable Posible

(c) El complementario de L_1 no es recursivamente enumerable, pero el complementario de L_2 es recursivamente enumerable Falso

(d) El complementario de L_2 no es recursivo, pero el complementario de L_3 es recursivo Falso

(e) Si L_1 es recursivo, entonces el complementario de L_2 es recursivo Falso

(f) Si L_3 es recursivo, entonces el complementario de L_4 es recursivo Verdadero

(g) Si L_3 es recursivamente enumerable, entonces la unión de L_2 y L_4 es recursivamente enumerable. Verdadero

(h) Si L_3 es recursivamente enumerable, entonces la intesección de L_2 y L_4 es recursivamente enumerable Verdadero

4 a) Conjunto cuadrados perfectos

1) Sumamos 1 en 2 cintas → Impares / → Cuadrados perfectos 

2) Colocamos suma en la cinta de salida

3) Ponemos separador

4) Sumamos 2 en la cinta de los impares

5) Sumamos impar último número colocado

6) Volvemos al paso 2

b) Conjunto de los naturales primos

- 1ª Cinta: Divisor
- 2ª Cinta: Dividendo
- 3ª Cinta: Número natural
- 4ª Cinta: Salida

1) Inicializamos sumando al dividendo y nº natural una unidad.

2) Sumamos una unidad al divisor

3) Comparamos que sea igual dividendo y divisor

4) Caso afirmativo: Se añade a la salida

5) Caso negativo: Se divide y vemos si es exacta

6) En caso de ser exacta, pasamos al paso 1.

7) En caso de no ser exacta, volvemos al paso 2.

18. Determinar cuáles de los siguientes problemas son decidibles, semidecidibles o no semidecidibles:

(a) Saber si una MT para una entrada 0011 no va a usar más de 10 casillas de la cinta.

(b) Dadas dos MT saber si aceptan el mismo lenguaje

a) ALG(M)

- Pongo a simular paso a paso M con 0011.
- En cada paso:
  - Si he usado más de 10 casillas, escribo Sí.
  - Si termino y no he usado más de 10 casillas, escribo No.
  - Compruebo si la configuración actual ha salido antes, escribo No. En otro caso, recuerdo configuración y paso a simular el siguiente paso.

b) No es semidecidible (Dos MT aceptan el mismo lenguaje si para toda palabra que acepte la primera, la segunda también lo acepta y viceversa).

VACIO(M) $\propto$ MISMOLENGUAJE(M_1, M_2)

$M \longmapsto (M_1, M_2)$

VACIO(M)

$M_1 \leftarrow M$

$M_2 \leftarrow M\emptyset$

RETURN MISMOLENGUAJE(M_1, M_2)

19. Determinar cuáles de los siguientes problemas son decidibles, semidecidibles o no semidecidibles:

(a) Dada una MT M y una palabra u, saber si la MT acepta la palabra u en un número de pasos menor o igual a $|u|$.

(b) Dada una MT determinar si no acepta la palabra vacía.

a) Decidible. Guardo $|u|$ en una variable (una cinta) y simulo la MT. Vamos contando los pasos y ver si el número de pasos que da es menor o mayor que $|u|$.

b) No semidecidible. Se puede "no aceptar" ciclando, lo cual no se puede comprobar en tiempo finito. (*)

El complementario (aceptar la palabra vacía) es semidecidible. Si este fuera también semidecidible, entonces sería decidible, lo cual no es posible por (*).

20. Determinar cuales de los siguientes problemas son decidibles, semidecidibles o no semidecidibles (se supone que las MTs tienen a $\{0, 1\}$ como alfabeto de entrada):

(a) Dadas dos máquinas de Turing, M_1 y M_2, determinar si existe, al menos, una palabra aceptada simultáneamente por ambas máquinas.

(b) (MANUAL) Dada una MT, determinar si para toda palabra de entrada no realiza más de 5 movimientos.

(c) Determinar si una MT no es determinista.

(d) (MANUAL) Dada una MT, determinar si ninguna de las palabras que acepta es un palíndromo.

4

a) Elegimos una palabra de forma no determinista, entonces puedo comprobar que si M_1 lo acepta, M_2 también.

C-VACIO(M) $\propto$ ALM1(M_1, M_2)

C-VACIO(M)

$M_1 \leftarrow M$

$M_2 \leftarrow M_T$ (Máquina de Turing que acepte todo)

RETURN ALM1(M_1, M_2)

c) Decidible

No nos preguntan nada sobre el funcionamiento de la MT, sino de su forma, lo cual suele ser decidible

22. Determinar, justificando las respuestas, cuáles de los siguientes problemas son decidibles, semidecidibles o no semidecidibles:

(a) Dada una máquina de Turing y una palabra de entrada determinar si visita menos de 10 casillas distintas de la cinta de lectura. Suponemos que solo hay 1 cinta

(b) Dadas dos Máquinas de Turing, determinar si el conjunto de las palabras aceptadas a la vez por ambas máquinas de Turing es finito.

a) Decidible (Igual que el 18 a))

b) No semidecidible

FINITO(M) $\propto$ INTERSECCIONFINITA(M_1, M_2)

FINITO(M)

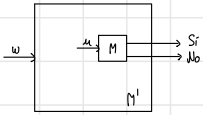
$M_1 \leftarrow M$

$M_2 \leftarrow M_T$ (Máquina de Turing que acepte todo)

RETURN INTERSECCIONFINITA(M_1, M_2)

•) Por el Teorema de Rice, FINITO(M) no es decidible.

•) C-UNIVERSAL $(M, u) \propto$ FINITO(M')



Luego, FINITO(M) no es semidecidible $\Rightarrow$ INTERSECCIONFINITA(M_1, M_2) no es semidecidible

Sí $\rightarrow$ M no acepta $u \rightarrow L(M') = \emptyset \rightarrow$ Sí (es finito)

No $\rightarrow$ M acepta $u \rightarrow L(M') = A^* \rightarrow$ No (no es finito)

23. Indica cuáles de los siguientes problemas son decidibles, semidecidibles o no semidecidibles:

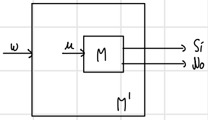
(a) Determinar si el lenguaje aceptado por una MT es regular.

(b) Dadas dos MTs determinar si hay una palabra que es aceptada por las dos MT y además es un palíndromo.

a) No es decidible (Teorema de Rice)

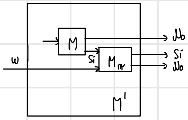
No es semidecidible

C-UNIVERSAL$(M, u) \propto$ REGULAR(M')



Sí $\rightarrow$ M no acepta $u \rightarrow L(M') = \emptyset \rightarrow$ Sí (es regular)

No $\rightarrow$ M acepta $u \rightarrow L(M') = A^* \rightarrow$ Sí (es regular) $\leftarrow$ Esta reducción no nos sirve



Sí $\rightarrow$ M no acepta $u \rightarrow L(M') = \emptyset \rightarrow$ Sí (es regular)

No $\rightarrow$ M acepta $u \rightarrow L(M') = L_{ne} \rightarrow$ No (no es regular)

No regular

24. Indica cuáles de los siguientes problemas son decidibles, semidecidibles o no semidecidibles:

(a) Dada una MT determinar si todos sus estados son accesibles (se puede llegar a ellos para un cálculo para alguna entrada).

(b) Dada una MT determinar si su número de estados es menor o igual a 5.

a) Es lo mismo que saber si el lenguaje aceptado es no vacío

(Pasamos en un primer paso a la izquierda y a un estado q_0', vamos pasando por los estados no finales y cuando hayamos terminado volvemos a la derecha y simulamos de normal la máquina de Turing
\+ Unificamos todos los estados finales en uno solo)

No es decidible

b) Es decidible

25. Demuestra que el Problema de la Correspondencia de Post con un alfabeto A que tiene un sólo elemento es decidible.

$$\boxed{\begin{array}{c} | \\ \hline ||| \end{array}} \quad \boxed{\begin{array}{c} ||| \\ \hline || \end{array}} \quad \boxed{\begin{array}{c} ||| \\ \hline || \end{array}} \quad \boxed{\begin{array}{c} | \\ \hline ||| \end{array}} \quad \boxed{\begin{array}{c} ||| \\ \hline || \end{array}}$$