

IIC 2213 – Lógica para ciencia de la Computación
Tarea 3 - Entrega Lunes 19 de Abril a las 20:00 - via siding

Recuerda que esta tarea es individual. Puedes discutir sobre la respuesta con tus compañeros (¡y eso está muy bien!), pero no puedes enviar la respuesta o utilizar la respuesta de alguien más. Esta tarea toca temas de computabilidad que son los pilares de la parte de complejidad de este curso. ¡Aprovecha de aprenderlo ahora!

Vamos a partir con algunas definiciones.

Cambios Un *cambio* en una ejecución de una máquina de turing es un punto en donde la máquina cambia de dirección. Específicamente, sea $E = c_1, \dots, c_n$ una ejecución de una máquina M con input w (en particular cada c_i es una configuración, y puedo ir de un paso desde cada c_i a c_{i+1}). Hay un cambio en E cuando hay configuraciones seguidas c_j, c_{j+1} y c_{j+2} , en el paso de c_j a c_{j+1} la máquina se movió en una dirección, y en el paso de c_{j+1} a c_{j+2} la máquina se movió en la otra dirección.

Conjunto Oscuro Sea $G = (V, E)$ un grafo no-dirigido. Un conjunto oscuro en G es un subconjunto $O \subseteq V$ de nodos tales que G no tiene ninguna arista entre ningún par de nodos de O (en otras palabras, $\{o_1, o_2\}$ no pertenece a E para cada par de nodos o_1 y o_2 en O).

Codificar Grafos Para trabajar con grafos vamos a usar la siguiente codificación $C(G)$ de grafos. Sea $G = (V, E)$ un grafo y $n = |V|$ su cantidad de nodos. Entonces $C(G)$ es el string $1^n \# w_1 \# w_2 \# \dots \# w_n$, donde cada w_i es un string binario que tiene un 1 en la posición j si $\{i, j\}$ es una arista en E , y un 0 en otro caso.

Por ejemplo, considera el grafo “cuadrado” con $V = \{1, 2, 3, 4\}$ y $E = \{\{1, 2\}, \{2, 3\}, \{3, 4\}, \{4, 1\}\}$. Nota que tanto $\{2, 4\}$ y $\{1, 3\}$ son conjuntos oscuros de nodos. la codificación de G se genera tomando $w_1 = 0101$, $w_2 = 1010$, $w_3 = 0101$ y $w_4 = 1010$, y luego

$$C(G) = 1111\#0101\#1010\#0101\#1010$$

Problema

- a) Piensa en lo siguiente. Tienes un grafo $G = (V, E)$, y asumes que $V = \{1, \dots, n\}$. Tienes también un string binario de n dígitos $a_1 \dots a_n$. Este string lo podemos usar para codificar un subconjunto S de V : cuando $a_i = 1$ significa que $i \in S$, cuando $a_i = 0$ significa que $i \notin S$. Considera ahora el lenguaje de todos los strings de la forma $a_1 \dots a_n \# C(G)$, tal que (1) G tiene n nodos, y tal que (2) G tenga un conjunto oscuro formado específicamente por todos los nodos i tal que $a_i = 1$. Por ejemplo, en el caso de que G sea el grafo cuadrado descrito arriba, la palabra $0101 \# C(G)$ sí debería pertenecer al lenguaje, por que $\{2, 4\}$ es un conjunto oscuro en G , pero $1100 \# C(G)$ no pertenece por que $\{1, 2\}$ no es un conjunto oscuro en G .

Muestra cómo aceptar este lenguaje con una máquina de turing determinista (puedes usar símbolos adicionales). Además, se requiere lo siguiente: por cada string $a_1 \cdots a_n \# C(G)$ que acepta tu máquina, lo hace con una ejecución que hace una cantidad de cambios en $O(n)$ (digamos una cantidad $d_1 n + d_2$ de cambios, con d_1 y d_2 constantes que no dependen ni de G ni de n). *Idea: Puedes ir bit por bit en $a_1 \cdots a_n$ y modificando los w_i s de forma que después puedas chequear los w_i s para ver si es un conjunto oscuro o no.*

- b) Explica como construir una máquina de turing no-determinista que acepte el lenguaje de todos los strings que son de la forma $1^k \# C(G)$ tal que G tiene un conjunto oscuro de tamaño k , y tal que para cada palabra $1^k \# C(G)$ que acepta tu máquina, con G un grafo de n nodos, lo hace con una ejecución que hace $O(n)$ cambios (digamos una cantidad $d_1 n + d_2$ de cambios, con d_1 y d_2 constantes que no dependen ni de k ni de G ni de n). *Idea: como puedes no-deterministicamente ir desde el string $1^k \# C(G)$ a algun string de la forma $a_1 \cdots a_n \# C(G)$ que ya sabes como aceptar por el item de arriba?*

Formato de entrega Un solo archivo pdf. Aceptamos documentos pdf escrito en latex, o, excepcionalmente, imágenes escaneadas o fotografiadas en buena calidad. Puedes resolver la b asumiendo que ya tienes la parte a), con eso obtienes un 4 en la tarea.