FUNDAMENTOS DE TEORÍA DE LA COMPUTACIÓN. COMPLEJIDAD TEMPORAL.
Trabajo Práctico Nro 5 Año 2020. Licenciatura en Sistemas.

Comentario: Hacer mínimamente los ejercicios 1 al 4. El ejercicio 5 es un poco más difícil, pero no debería haber mayores problemas para resolverlo, con lo que vimos en la clase nro 5.

Ejercicio 1. Responder y justificar brevemente las siguientes preguntas conceptuales:

a) ¿Por qué sólo tiene sentido tratar la complejidad temporal dentro de la clase R?

b) Resolvimos de dos maneras el problema de los palíndromos, una con una MT de 1 cinta y otra con una MT con varias cintas. La primera tarda $O(n^2)$ pasos y la segunda $O(n)$. Al igual que para otros problemas que manifiestan este comportamiento, ¿por qué es indistinta la cantidad de cintas utilizadas, considerando la jerarquía temporal que definimos?

c) Probar, utilizando la definición vista en clase, que $n^3 = O(2^n)$.

d) Vimos que un algoritmo natural para encontrar un divisor que termine en 3 de un número N tarda O(N) pasos. ¿Esto significa que el problema está en P?

e) Vimos que utilizando una MTN (MT no determinística), un algoritmo natural para encontrar un circuito de Hamilton en un grafo G tarda $O(n^2)$ pasos. ¿Esto significa que el problema está en P?

f) El problema de los grafos isomorfos se representa por el lenguaje ISO = $\{(G_1, G_2) \mid G_1$ y G_2 son grafos isomorfos$\}$. Dos grafos son isomorfos si son idénticos salvo por la disposición de sus vértices. P.ej. el cuadrado con arcos (1,2), (2,3), (3,4) y (4,1) es isomorfo al cuadrado con arcos (1,2), (2,4), (4,3) y (3,1). Se prueba que ISO $\in$ NP. Mostrar un certificado suscinto que caracterice al problema ISO. *Ayuda: Notar en el ejemplo, con $G_1 = (V_1,E_1)$ y $G_2 = (V_2,E_2)$, que si π es la permutación de V_1 siguiente: $\pi(1) = 1$, $\pi(2) = 2$, $\pi(3) = 4$, $\pi(4) = 3$, entonces E_1 coincide con E_2 aplicando sobre los vértices de este último la permutación π.*

g) Ligado al inciso anterior, considerando ahora el problema de los grafos no isomorfos, se pide mostrar un certificado asociado al problema. ¿Es suscinto?

h) Probar que la clase P es cerrada con respecto a la operación de complemento.

Ejercicio 2. Probar que si $T_1(n) = O(T_2(n))$, entonces $\text{TIME}(T_1(n)) \subseteq \text{TIME}(T_2(n))$. *Ayuda: Usar las definiciones estudiadas para probar la inclusión entre los conjuntos indicados.*

Ejercicio 3. Sea el lenguaje SMALL-SAT = $\{\varphi \mid \varphi$ es una fórmula booleana sin cuantificadores en la forma normal conjuntiva (o FNC), y existe una asignación de valores de verdad que la satisface en la que hay a lo sumo 3 variables con valor de verdad verdadero$\}$. Probar que SMALL-SAT $\in$ P. *Comentario: φ está en la forma FNC si es una conjunción de disyunciones de variables o variables negadas, como p.ej. $(x_1 \vee x_2) \wedge x_4 \wedge (\neg x_3 \vee x_5 \vee x_6)$.*

Ejercicio 4. El problema del conjunto dominante de un grafo se representa por el lenguaje DOM-SET = $\{(G, K) \mid G$ es un grafo que tiene un conjunto dominante de K vértices$\}$. Un subconjunto de vértices de un grafo G es un conjunto dominante de G, si todo otro vértice de G es adyacente a algún vértice de dicho subconjunto. Probar que DOM-SET $\in$ NP. ¿Se cumple que DOM-SET $\in$ P? ¿Se cumple que DOM-SET$^C \in$ NP? Justificar las respuestas.

Ejercicio 5. Sea M una MTN que si acepta una cadena w, al menos en una de sus computaciones lo hace en tiempo polinomial con respecto a |w|, digamos p(|w|). Probar que L(M) $\in$ NP. *Ayuda: Se sabe que toda función polinomial p(n) es tiempo-construible, es decir que se computa en tiempo p(n).*