Complejidad Computacional. Tarea 1

Semestre 2019-II

1. Encuentra y demuestra las relaciones asintóticas en términos de la notación O entre los siguientes pares de funciones.
   (a) $f(n) = a_k n^k$, $g(n) = \sum_{i=0}^{k} a_i n^i$, donde $a_1, \ldots a_k$ son constantes.
   (b) $f(n) = n^3$, $g(n) = n^3\sqrt{n}$.
   (c) $f(n) = \sqrt[3]{n}$, $g(n) = \sqrt[2]{n}$.
   (d) $f(n) = \log_2(n)$, $g(n) = \log_3(n)$.
   (e) $f(n) = \log_2(n^3)$, $g(n) = \log_2(n^2)$.
   (f) $f(n) = n\log_3(n)$, $g(n) = n\log_2(n)$.

2. Define una máquina de Turing que decida el siguiente lenguaje:
$$L = \{x \in \{0,1\}^* : \exists n \in \mathbb{N}. |x| = 2^n\}.$$
   Encuentra el tiempo de ejecución $T(n)$ de esta máquina y el orden de acuerdo a O a cual que éste pertenece.

3. Sea $c \in \mathbb{N}$ una constante. Sea M una máquina de Turing que con función de transición
$$\delta : Q^k \times \Gamma \to Q^k \times \Gamma \times \{\to, \leftarrow\},$$
   donde Q^k indica que la función de transición acepta k estados, que computa a f en tiempo $T(n)$. De muestra que existe una máquina de Turing M' con función de transición
$$\delta' : Q' \times \Gamma \to Q' \times \Gamma \times \{\to, \leftarrow\}$$
   que computa a f en tiempo $O(T(n))$.

4. Definamos a una máquina de Turing con memoria RAM como aquella que tiene memoria de acceso aleatorio. Formalmente, la máquina cuenta con dos símbolos adicionales R y W, un estado adicional llamado q_{access}, una cinta adicional llamada *cinta de dirección* y una memoria en forma de un arreglo infinito A inicializado en blanco. Cada que esta máquina entra a q_{access}, si la cinta de dirección contiene la cadena $\llcorner i \lrcorner R$ (donde $\llcorner i \lrcorner$ es el número i en notación binaria) el contenido de $A[i]$ (el cuál es un símbolo en el alfabeto)

es escrito en la cinta de dirección después del símbolo R. Si la cinta de dirección contiene $\llcorner i \lrcorner W\sigma$ (con $\sigma \in \Sigma$) entonces guardamos σ en $A[i]$.

Demuestra que si una función Booleana $f : \{0,1\}^* \rightarrow \{0,1\}$ es computable en tiempo $T(n)$ por una máquina de Turing con RAM, entonces $f \in \mathbf{DTIME}(T(n)^2)$.

5. Un número $n \in \mathbb{N}$ está representado por la cadena binaria $d_{\log n} \dots d_0 d_1$ sii

$$n = \sum_{i=0}^{n} d_i \times 2^i.$$

En general, si tomamos como *base* el número b y tenemos el alfabeto $\Sigma = \{0, \dots, b-1\}$, podemos decir que una cadena $d_{\log n} \dots d_0 d_1 \in \Sigma^*$ representa a n sii $d_{\log n} \dots d_0 d_1$ sii

$$n = \sum_{i=0}^{n} b_i \times 2^i.$$

Denotemos por $\llcorner n \lrcorner_b$ a la representación del número n en base b, $\llcorner n \lrcorner_b \in \{0, \dots, b-1\}^*$.

Sean $S \subseteq \mathbb{N}$ y $L_S^b = \{\llcorner n \lrcorner_b : n \in S\}$, **demuestra que**

$$L_S^b \in \mathbf{P} \iff L_S^2 \in \mathbf{P}.$$

Ésta es una manera formal de decir que una base distinta para representar los números no tiene ninguna ningún efecto sobre su pertenencia a $\mathbf{P}$.

6. Demuestra que los siguientes lenguajes están en $\mathbf{P}$ (elige la representación que prefieras para las gráficas):

   - CONNECTED: el conjunto de todas las gráficas conexas.
   - BIPARTITE: el conjunto de todas las gráficas bipartitas, es decir, es decir, aquellas cuyos vértices puedan ser divididos en dos conjuntos A y B tales que todas las aristas en la gráfica tengan en un extremo un vértice de A y en el otro uno de B.

7. Supón que $L_1, L_2 \in NP$. ¿Qué pasa con $L_1 \cup L_2$ y $L_1 \cap L_2$? Demuestra tus conclusiones.

8. Demuestra que si $\mathbf{P} = \mathbf{NP}$, entonces $\mathbf{NP} = \mathbf{coNP}$.