Ma50b Complejidad Computacional **10 de Junio de 2010**

Control 2

Prof. Cátedra: M. Kiwi *Prof. Auxiliar: R. Briceño*

TIEMPO: 5.0 HRS.

PROBLEMA 1:

(i).- (2.0 pts) Se define el lenguaje DS (por *dominating set*) como la colección de $\langle G,k\rangle$ donde $G=(V,E)$ es un grafo, $k\in\mathbb{N}$, y existe $D\subseteq V$ conjunto de nodos tal que $|D|\leq k$ y donde todo nodo $u\in V\setminus D$ esta dominado por D, i.e. existe un nodo $v\in D$ tal que $vu\in E$. Pruebe que DS es NP-completo.

Indicación: Reduzca desde un lenguaje relacionado con grafos.

(ii).- (2.0 pts) Dado un circuito Booleano $C(x_1,\ldots,x_n,y_1,\ldots,y_n)$ que toma como entrada dos vectores $\vec{x}=(x_1,\ldots,x_n),\vec{y}=(y_1,\ldots,y_n)\in\{0,1\}^n$, decimos que C tiene un *ciclo de entrada* si existe algún $m\in\mathbb{N}$ y $\omega_0=0^n,\omega_1,\ldots,\omega_{m+1}=0^n$ tal que $C(\omega_i,\omega_{i+1})=1$ para todo $0\leq i\leq m$.

Se define Ciclo como el conjunto de palabras $\langle C\rangle$ donde C es un circuito como el descrito en el párrafo precedente y tal que C tiene un ciclo de entrada. Pruebe que Ciclo $\in$ PESPACIO.

(iii).- (2.0 pts) Se sabe que DESPACIO$(n)\neq$ PESPACIO. Use este hecho para probar que DESPACIO$(n)\neq$ P.

Indicación: Use una argumento de padding.

PROBLEMA 2:

(i).- (2.0 pts) Sea P^{FBTC} la clase de lenguajes decididos por máquinas de Turing a tiempo polinomial con oráculo $FBTC$. Sea NP^{FBTC} la clase de lenguajes decididos por máquinas de Turing no–deterministas a tiempo polinomial con oráculo $FBTC$.[1]

Pruebe que PESPACIO $\subseteq \mathrm{P}^{FBTC}\subseteq \mathrm{NP}^{FBTC}\subseteq$ PESPACIO, y concluya que $\mathrm{P}^{FBTC}=\mathrm{NP}^{FBTC}$.

(ii).- El objetivo de esta parte es probar que existe un oráculo A tal que $\mathrm{P}^A\neq\mathrm{NP}^A$.

> (ii.1).- (0.75 pts) Sea $L_A=\{0^n : \exists x\in A, |x|=n\}$. Pruebe que para todo oráculo A se tiene que $L_A\in\mathrm{NP}^A$.

Sea $M_1^?,M_2^?,\ldots$ una enumeración de todas las máquinas de Turing con acceso a un oráculo y a tiempo polinomial. Dado que hay una infinidad de maneras de codificar un máquina de Turing agregando estados inútiles, sin pérdida de generalidad se puede asumir que cada máquina aparece una infinidad de veces en la enumeración.

[1] Sea L un lenguaje. Una máquina de Turing con oráculo L es una máquina de Turing multicintas, una de las cuales se denomina cinta de pregunta. Además, la máquina de Turing tiene tres estados especiales $q_?$, q_s, y q_n. El estado $q_?$ se utiliza para preguntar si el contenido de la cinta de pregunta está en L. La respuesta se obtiene en la siguiente transición al entrar la máquina en uno de los estados q_s o q_n dependiendo de si la respuesta es *si* o *no*. Luego, la máquina sigue iterando hasta la próxima vez que entre en el estado $q_?$. La máquina de turing M con oráculo L se denota M^L.

(ii.2).- (0.75 pts) Justifique que cualquiera sea el oráculo A, se puede asumir que $M_i^?$ es a tiempo $(n+2)^i$ sin que esto cambie la clase de lenguajes $\mathcal{C}_A = \left\{L(M_i^A) : i \geq 1\right\}$.

Ahora veremos como definir A. Inicialmente $A_0 = X = \emptyset$. Supongamos que se ha definido A_{i-1} para $i \geq 1$. Sea $n_i' = \text{máx}\{|\omega| : \omega \in A_{i-1}\}$ y $n_i^{''}$ el menor entero n tal que $2^n > (n+2)^i$. Definimos $n_i = \text{máx}\{n_i^{'}, n_i^{''}\} + 1$. Se simula $M_i^?$ en 0^{n_i}. Supongamos que durante la simulación se pregunta al oráculo por la membrecia en él de ω. Si $|\omega| < n_i$, entonces se prosigue la simulación tomando como respuesta *si* o *no* dependiendo de si ω pertenece o no a A_{i-1}. Si $|\omega| \geq n_i$, entonces se prosigue la simulación asumiendo que la respuesta fue *no* y se agrega ω al conjunto X. Si $M_i^?$ acepta 0^{n_i}, hacemos $A_i = A_{i-1}$. En caso contrario, hacemos

$$A_i \quad = \quad A_{i-1} \cup \{\omega \in \{0,1\}^* : |\omega| = n_i, \omega \notin X\} \ .$$

Finalmente, definimos $A = \cup_{i \in \mathbb{N}} A_i$.

(ii.3).- (1.25 pts) Pruebe que $M_i^{A_{i-1}}$ acepta 0^{n_i} si y solo si M_i^A acepta 0^{n_i}.

(ii.4).- (1.25 pts) Pruebe que $L_A \notin \mathrm{P}^A$.