

INF155: Informática Teórica

Clase 10: Un Lenguaje que no es regular

Aldo Berrios Valenzuela

Jueves 7 de Abril de 2016

1. Un lenguaje que cumple LB sin ser regular

Teorema 1.1. El lenguaje $L = \{ab^n c^n : n \geq 0\} \cup \{a^i b^j c^k : i \neq 1 \wedge j \geq 0 \wedge k \geq 0\}$ cumple el lema de bombeo, pero no es regular.

Demostración. La conclusión del Lema del Bombeo asegura que toda palabra $\sigma \in L$ con $|\sigma| \geq N$ (suponiendo que L es un lenguaje regular) se puede particionar en $\sigma = \alpha\beta\gamma$ tal que $|\beta| > 0$, $|\alpha\beta| \leq N$ y $\alpha\beta^k\gamma \in L$.

i. σ comienza con a :

Si bombeamos a , queda $a^k b^n c^n \in L$.

Si $\sigma = a^r b^s c^k$ con $r \geq 2$, al bombear la a queda $a^{r+k-1} b^s c^k \in L$, porque al ser $k \geq 0$, $r \geq 2$ queda $r+k-1 \geq 1$.

ii. σ comienza con b o c : No hay problema en bombear, queda algo en $b^* c^* \in L$

\rightsquigarrow cumple las conclusiones del lema de bombeo.

Pero L no es regular. Consideremos el homomorfismo:

- $h(a) = \epsilon$
- $h(b) = a$
- $h(c) = b$

Consideremos:

$$h(L \cap \mathcal{L}(ab^*c^*)) = \{a^n b^n : n \geq 0\}$$

que sabemos que no es regular. Si L fuera regular, esto sería regular. Entonces, por contradicción se tiene que L no es regular. \square

1.1. Estrategia de demostración del LB

Estrategia de lema de bombeo para demostrar lenguaje no es regular: juego con un adversario, que elige los peores casos para \exists , uno elige a gusto para \forall (Cuadro 1).

Sansano	Adversario
Propone L	
Elige $\sigma \in L, \sigma \geq N$	Pone N , constante del lema
Elige $k = 0$ o $k \geq 2$ tal que $\alpha\beta^k\gamma \notin L$	Divide $\sigma = \alpha\beta\gamma, \alpha\beta \leq N, \beta \neq \epsilon$
	#\$!@5 sansano, me ganó de nuevo... ya verá la próxima

Cuadro 1: Estrategia para demostraciones con lema de bombeo. En esta ocasión, presentamos la intensa pelea entre el sansano y su vil adversario.

Ejemplo 1.1. $L = \{a^p : p \text{ primo}\}$ no es regular.

Demostración. Elegimos $\sigma = a^p$, con $p \leq N$ primo. Basta considerar largos:

$$|\beta| = u, \quad 1 \leq u \leq N \leq p$$

Por el lema de bombeo, es primo:

$$p + (k-1)u$$

para todo $k \geq 0$.

Si elegimos $k = p + 1$, es primo:

$$p + (p+1-1)u = p(1+u)$$

y $u+1 \geq 2$. Entonces, por contradicción se tiene que L no es regular. \square

EQVA Sea L un lenguaje que está compuesto por todos los factoriales de un número binario. Demuestre que no es regular.

Hint: considere los largos.

2. Autómatas de Stack (PDA)

Idea: NFA + Stack como memoria auxiliar.

Definición 2.1. Un PDA $M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ consta de:

- Q : conjunto finito de estados.
- Σ : alfabeto de entrada.
- Γ : Alfabeto de stack.
- δ : función de transición:

$$\delta : Q \times (\Sigma \cup \{\epsilon\}) \times \Gamma \rightarrow \underbrace{2^{Q \times \Gamma^*}}_{\text{subconjuntos finitos}}$$

- q_0 : estado inicial, $q_0 \in Q$.
- Z_0 : stack inicial, $Z_0 \in \Gamma$
- F : estados finales, $F \subseteq Q$.

Idea: Si M está en el estado q , y en el tope del stack tiene A , al leer x ($x \in \Sigma$, o $x = \epsilon$) de la entrada, si $(p, \gamma) \in \delta(q, x, A)$ entonces M puede pasar al estado p , consume x de la entrada, y reemplaza A por γ en el stack.

Con dibujitos:

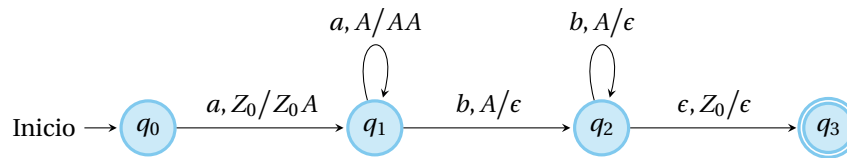


Figura 1: Ejemplo de un PDA en forma de digrafo.

Los arcos de un PDA están sujetos a la notación

$$x, A/B \quad (2.1)$$

donde x es un símbolo, A y B son símbolos del stack. La notación (2.1) se lee como: “si leo el caracter x y mi tope del stack es A , avanzo al siguiente estado reemplazando el tope del stack A por B ”. Por ejemplo, para el PDA de la Figura 2, si estamos en el estado q_0 podremos acceder a q_1 únicamente si leemos el símbolo a y al mismo tiempo,

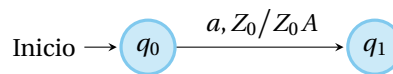


Figura 2: Otro PDA

si Z_0 es el tope del stack. En ese caso pasamos a q_1 consumiendo a y reemplazando el tope del stack (que en ese momento es Z_0) por Z_0A .