

# PRÁCTICO 6 LENGUAJES FORMALES:

## Máquinas de Turing

Mauricio Velasco

1.
  - a) Diseñe y escriba una máquina de Turing que escanea hacia la derecha hasta que encuentra dos  $a$ 's consecutivas y luego se detiene. El alfabeto de la máquina debe ser  $\Sigma = \{a, b, \cup, \Delta\}$  y debe dar la descripción de la máquina en completo detalle (como tupla).
  - b) Escriba las configuraciones que ocurren al ejecutar su máquina con input  $\cup bbabaa$ .
2. Construya una máquina de Turing (usando nuestra notación abreviada) que calcule la función  $f : \{a, b\}^* \rightarrow \{a, b\}^*$  dada por  $f(w) = ww^R$  donde  $w^R$  significa la palabra reversa a  $w$ . Muestre la ejecución de la misma en una cadena representativa.
3. Describa una máquina de Turing que semidecida el lenguaje  $a^*ba^*b$ .
4. Utilice máquinas de Turing no deterministas para demostrar que:
  - a) La clase de lenguajes recursivos está cerrada bajo unión, concatenación y estrella de Kleene.
  - b) La clase de lenguajes recursivamente enumerables está cerrada bajo unión, concatenación y estrella de Kleene.
5. (*Autómatas con dos stacks*)
  - a) Defina formalmente un automata que sea un *pushdown automata con dos stacks*, especificando definición, configuraciones y cómputo. Defina formalmente lo que significa que esta máquina *acepte* un lenguaje.
  - b) Demuestre que un lenguaje es recursivo si y solo si es aceptado por un *pushdown automata con dos stacks*.
6. Encuentre gramáticas que generen los siguientes lenguajes:

a)  $L = \{ww : w \in \{a, b\}^*\}$

b)  $L = \{a^{2^n} : n \in \mathbb{N}\}$

c)  $L = \{a^{n^2} : n \in \mathbb{N}\}$