

Autómatas y Lenguajes Formales. Tarea 2.1

1. Minimiza los autómatas de tus respuestas a los ejercicios 4 de la tarea 1.1 y 2 de la tarea 1.2
2. Construye el autómata de diccionario para el conjunto $X = \{ab, ba, aba, bab\}$ con el alfabeto $\Sigma = \{a, b\}$.

Considera la siguiente descripción de un lenguaje de programación simple:

- Localidades de memoria: X, Y, Z, X_1, \dots ;
 - Constantes: $0, 1, -1, \dots$;
 - Expresiones aritméticas: (a) localidades; (b) constantes; (c) si a y b son expresiones aritméticas, también lo son $(a + b)$, $(a \times b)$ y $(a - b)$;
 - Constantes booleanas: V y F ;
 - Comparaciones: $(X = a)$ y $(X < a)$, donde X es una localidad y a una expresión aritmética;
 - Expresiones booleanas: (a) constantes booleanas; (b) comparaciones; (c) si b y v son expresiones booleanas, también lo son $\neg b$, $(b \vee v)$ y $(b \wedge v)$;
 - Asignaciones: $X := a$, donde X es una localidad y a una expresión aritmética;
 - El programa skip;
 - Programas: (a) skip; (b) asignaciones; (c) si P y Q son programas y b es una expresión booleana, los siguientes también son programas: $(P; Q)$, $(\text{if } b \text{ then } P \text{ else } Q)$ y $(\text{while } b \text{ do } P)$.
3. Da una gramática libre de contexto que genere todos los programas en este lenguaje. No te preocupes por el formato de la gramática, simplemente trata de dar el menor número de reglas posibles.
 4. Da gramáticas en forma normal de Chomsky y de Greibach del mismo lenguaje.
 5. Describe un NPDA que acepte este lenguaje de programación.