

AUTÓMATAS Y LENGUAJES

PRÁCTICA 2

Jesús Miguel Álvarez Domínguez y Diana Rojas López

Descripción

En esta práctica hemos desarrollado una librería en c para poder simular la definición de un autómata finito no determinista con transiciones λ .

Para implementar esta nueva funcionalidad, hemos añadido una nueva matriz de índices que representará estas transiciones lambda. Por decisión de diseño, esta matriz contendrá 1s en su diagonal (todo estado tiene una transición lambda consigo mismo) y en todas aquellas casillas en las que se haya añadido una transición lambda mediante la función *AFNDInsertaLTransicion*.

Además, si existe una sucesión de estados unidos por transiciones lambda, dichas transiciones también se representan en la matriz. Es decir, si q1 y q2 tienen una lambda, y q2 y q3 otra; q1 y q3 estarán relacionados también. Estas uniones transitivas se calculan en *AFNDCierraLTransicion*.

Cada vez que se procesa un nuevo símbolo de la cadena de entrada, se añaden a los estados activos todos aquellos con los que existen relaciones lambdas desde los estados activos actuales.

Pruebas

Según abrimos la carpeta principal de nuestra práctica, encontramos el makefile que compilaremos para probar nuestros módulos. En la carpeta lib tenemos todos los códigos de los diferentes módulos (tanto los ficheros .h como los .c)

Una vez que hacemos *make all-test* desde la carpeta raíz, se nos generan los ejecutables de los ficheros *nombremodulo-test.c* que están en /lib.

Con *./pruebas_afnd* se comprueba el funcionamiento del autómata expuesto en el enunciado de la práctica.