

UNIVERSIDAD AUTÓNOMA



PRÁCTICAS DE AUTLEN

PRÁCTICA 3

Memoria

Autores:

AITOR ARNAIZ DEL VAL

SANTIAGO GONZÁLEZ-CARVAJAL

CENTENERA

Pareja 9

Grupo 1401

13 de diciembre de 2018

Índice

1. Descripción de la implementación.	2
1.1. AFND1ODeSimbolo.	2
1.2. AFNDAAFND1O.	2
1.3. AFND1OUne.	2
1.4. AFND1OConcatena.	2
1.5. AFND1OConcatena.	2
1.6. AFNDADot.	2
2. Comentarios sobre la práctica y el formato.	3

1. Descripción de la implementación.

Nótese que todas las funciones han sido más sencillas de implementar gracias al uso de las funciones de las prácticas anteriores.

1.1. AFND1ODeSimbolo.

Estas tres funciones han sido bastante sencillas gracias al uso de las funciones hechas en las prácticas anteriores.

1.2. AFNDAAFND1O.

Esta era la función con mayor dificultad de la práctica, debido a que en ella primero teníamos que comprobar una serie de condiciones sobre los estados, y si no se cumplían aplicar el algoritmo indicado. Aplicar este algoritmo, debido al manejo de datos en C, ha sido de una dificultad considerable ya que teníamos que dar más memoria a los punteros y tener guardados todos los índices a editar con memoria dinámica. Nótese que esta es la única función **destructiva**, es decir, cambia el autómata que recibe como entrada. De ahí que luego la llamemos desde *AFND1OUne*.

1.3. AFND1OUne.

En esta función simplemente añadíamos todos los estados sin ningún cuidado para luego llamar a *AFNDAAFND1O* que corrige los problemas generados.

1.4. AFND1OConcatena.

Esta función es parecida a la de *AFNDAAFND1O* pero tiene cambios sutiles para generar la concatenación de los autómatas.

1.5. AFND1OConcatena.

Esta función ha sido la más sencilla de las anteriores.

1.6. AFNDADot.

Función que simplemente imprime en un fichero con el formato proporcionado en el enunciado de la práctica. Nótese que esta función hace uso de una variable global para generar dinámicamente el nombre del fichero de salida.

2. Comentarios sobre la práctica y el formato.

Lo primero, comentar que el main de prueba proporcionado en Moodle (el que se encuentra en el *.zip*, es decir *pr_er_enunciado.c*), no da la salida esperada dado que falta hacer el cierre de las transiciones lambda (que sí se hace en el main proporcionado en el *.pdf* con el enunciado de la práctica). Por ello, hemos creado un fichero de nombre *main.c* que parte del main del PDF y añade todas las pruebas del main del *.zip*, por ejemplo la prueba de la impresión a *.dot*.

Por último, comentar que de repente, la salida de la práctica 3 (estipulada en el *.zip*) cambia la salida correspondiente a la funcionalidad de la práctica 2. Concretamente, cambia la impresión de la matriz de lambdas suponiendo que, teóricamente, se ha seguido un algoritmo determinado para hacer los cierres reflexivo y transitivo (de potencias de la matriz) de manera que espera guardar todos los pasos del algoritmo en diferentes copias de la matriz para luego imprimirlos uno a uno. Por ello, y dado que era funcionalidad de la práctica 2 y no influye para nada en el funcionamiento del autómata, el cual es correcto y el esperado, hemos decidido no cambiar la impresión, dado que nuestro algoritmo para calcular los cierres no consta de esas tres fases identificadas, y se nos dio libertad para implementar los cierres con nuestro propio algoritmo.