

MEMORIA PRACTICA 2: AFD y AFND



Adrián Álvarez Barrera

Algorítmica y Modelos de Computación

INDICE

INTRODUCCIÓN

DEFINICIÓN DEL LENGUAJE

Características Generales

Requisitos del AFD

Requisitos del AFND

AUTÓMATAS: IMPLEMENTACIÓN

Implementación del AFD

Implementación del AFND

EJEMPLOS

EJEMPLO AFD

Ejemplo AFND

INTRODUCCIÓN

En esta práctica, la practica 2 de la asignatura, desarrollaremos un simulador de Autómatas Finitos Determinista (AFD) y de Autómatas Finitos No Deterministas (AFND) en Java, de forma que las transiciones del Autómata que pretendamos simular puedan ser dadas por teclado o leídas desde un fichero.

DEFINICIÓN DEL LENGUAJE

Características Generales

La serie de los autómatas, sus datos, están guardados en un fichero de texto plano en formato .tsp

Requisitos del AFD

Al principio se especifican una lista d estados, un estado inicial, una lista de transiciones y para finalizar una lista de estados finales. Todos estos estados, los estados iniciales y finales, se declaran en una sola linea.

Todos los estados de la lista, las transiciones, el estado inicial y el final, deben apararecer en la lista de estados.

Las transiciones se declaran en lineas continuas. La primera línea se encarga de identificar el comienzo de la lista, y la última se encarga de marcar su fin. En una línea solo puede haber una transición. Las transiciones están formadas por el estad del que se parte, separando por un espacio el símbolo que provoca esa transformación y separado por un espacio el estado al que se llega.

Un ejemplo de todo esto es:

```
ESTADOS: A B C
INICIAL: A
FINALES A B

TRANSICIONES:
  A '0' A
  A '1' B
  B '0' B
  B '1' A
FIN
```

Para que un AFD sea válido y pueda ejecutarse y hacerlo funcional, todos estos campos especificados anteriormente, han de ser obligatorios.

Requisitos del AFND

Los AFND necesitan una lista de transiciones lambda, además de una lista de estados inicial, una lista de estados finales y una lista de transiciones.

Estas transformaciones se definen de forma muy similar a las transiciones que consumen símbolos salvo que carecen de símbolo, solo es necesario el estado de partida y una lista de estados a los que se llega.

Un ejemplo de ello es:

```
TRANSICIONES LAMBDA:
  A B C
  B C
FIN
```

Las transiciones lambda no son un campo obligatorio, ya que para que un AFND sea correcto y las nuevas listas dichas, pueden estar formadas por un solo elemento. Es por eso que todo AFD puede ser un AFND, pero no al revés.

AUTÓMATAS: IMPLEMENTACIÓN

Implementación del AFD

Los estados iniciales y finales están representados con la clase String y los símbolos con la variable char.

La lista de estados iniciales y finales, se almacenan en un ArrayList.

El conjunto de estados no puede estar vacío, el estado inicial debe estar en el conjunto de estados. El conjunto de estados finales no pueden estar vacío tampoco. El estado inicial debe estar escrito en el conjunto de estados. El conjunto de estados finales no puede estar vacío, y todos sus elementos deben pertenecer al conjunto de estados.

El estado actual comienza siendo el primero, por cada símbolo de la cadena pasada, se comprueba si es válido o no, si no lo es, se produce una exception. si es válida, se procesa la transición, y así se reemplaza la del estado actual.

Cuando se procesa todo y se comprueba, la cadena se ejecuta.

Implementación del AFND

Los estados cambian, los estados iniciales siguen siendo estando representados con la clase String, pero los estados finales pasan a ser ArrayList. Los valores de las transiciones y de lambda se almacenan en ArrayList.

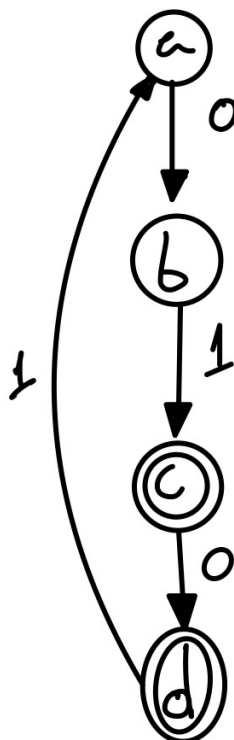
El proceso de los estados no cambia con respecto al AFD, solo que los estados finales tienen que estar incluidos en todos los estados.

Para calcular la lambda, se agregan todos los estados a los que se llega sin consumir símbolos de estado. Al introducir algún nuevo estado en la lista de nuevos estados, se examinan las lambdas transiciones de ese nuevo estado. Solo se agregan los estados que no estuviesen ya incluidos en el conjunto de nuevos estados para evitar bucles infinitos.

Una vez calculada lambda, se calculan las transiciones que consumen símbolos. Cuando se termina de buscar transiciones se vacía el macroestado y se agregan todos los nuevos.

EJEMPLOS

EJEMPLO AFD



ESTADOS: a b c d

INICIAL: a

FINALES: c d

TRANSICIONES:

a '0' b

b '1' c

c '0' d

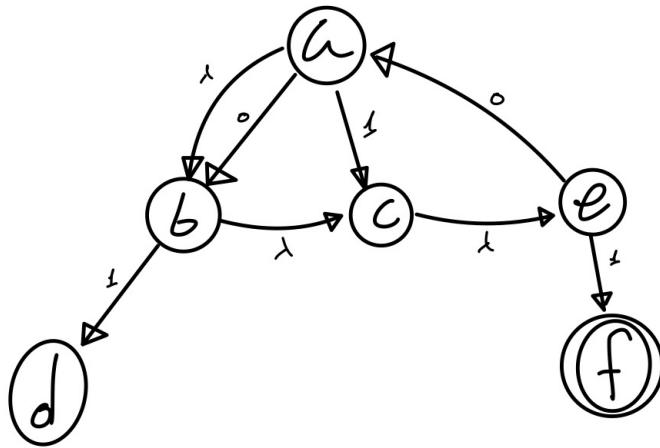
d '1' a

FIN

Cadena: 010101 -> Aceptada

Cadena: 0011 -> ERROR

Ejemplo AFND



ESTADOS: a b c d e f

INICIAL: b

FINALES: f

TRANSICIONES:

a '1' c

e '0' a

b '1' d

a '0' b

e '1' f

TRANSICIONES :

b c

a b

c e

FIN

Cadena: 001 -> Aceptada

Cadena: 111 -> ERROR