



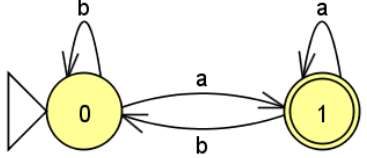
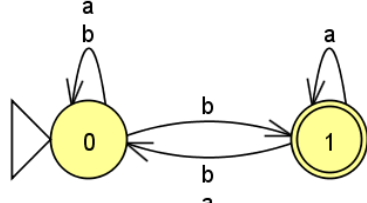
LENGUAJES FORMALES
Enunciado Práctica 2
Publicación: 1 Mayo 2018
Entrega: hasta 27 Mayo 2018
En grupos de tres

Ejecutor de Autómatas Finitos

Basándonos en el Analizador Léxico desarrollado en la primera práctica, realizar un traductor al lenguaje de programación C del formato de representación de AFs que viene descrito en el esquema siguiente:

```
% Autómata Finito
Alfabeto { símbolo , símbolo , ... }
Estados { num }
Transiciones { (num , símbolo ; num), ... }
Inicial { num }
Finales { num, ... }
```

Se pide, por tanto, modificar convenientemente el Analizador Léxico con FLEX y construir un Analizador Sintáctico con BISON que (1) dado un AF (determinista o indeterminista) construya una función `transición` en C que implemente la función de transición del AF. Ver tercera columna en tabla adjunta.

AF determinista		<pre>int transicion (int estado, char símbolo) { int sig; if ((estado==0)&&(símbolo=='a')) sig = 1; if ((estado==0)&&(símbolo =='b')) sig= 0; if ((estado==1)&&(símbolo =='a')) sig = 1; if ((estado==1)&&(símbolo =='b')) sig = 0; return (sig); }</pre>
AF no determinista		<pre>int * transicion (int estado, char símbolo) { static int sig[num-estados+1], n=0; if ((estado==0)&&(símbolo =='a')) sig[n++] = 0; if ((estado==0)&&(símbolo =='b')) sig[n++] = 0; if ((estado==0)&&(símbolo =='b')) sig[n++] = 1; if ((estado==1)&&(símbolo =='a')) sig[n++] = 1; if ((estado==1)&&(símbolo =='a')) sig[n++] = 0; if ((estado==1)&&(símbolo =='b')) sig[n++] = 1; sig[n] = -1; /* centinella */ return (sig); }</pre>

Observar que cada una de las transiciones `(num1 , símbolo ; num2)` del AF, dará lugar a una línea de código C de la forma `"if ((estado==num1)&&(símbolo==símbolo)) sig=num2;"` si el AF es determinista, o de la forma `"if ((estado==num1)&&(símbolo==símbolo)) sig[n++] = num2;"` si el AF es no determinista.

También se pide la gestión de errores y avisos (i.e., *warnings*) siguientes: (2) un `alfabeto` sin símbolos debe dar el error *"[ERROR] El alfabeto debe contener uno o más símbolos"*; (3) un AF con ningún o más de un estado `inicial` debe dar el error *"[ERROR] Los Autómatas Finitos solo deben tener un estado inicial"*; (4) un AF sin estados `finales` debe dar el error *"[ERROR] Los Autómatas Finitos deben tener algún estado final"*; (5) el uso de números de estado o símbolos inexistentes en una transición debe dar el error *"[ERROR] El estado %i de la transición (%i , %c ; %i) es desconocido"* o *"[ERROR] El símbolo %c de la transición (%i , %c ; %i) es desconocido"*, respectivamente. El proceso de compilación no debe detenerse, sino seguir procesando las `transiciones` y dar todos los errores seguidos. Si hay un error, no se genera el código de esa transición errónea.

Para los avisos o *warnings*: (6) si en la declaración del alfabeto hay un símbolo repetido, no tenerlo en cuenta y lanzar un aviso de *"[AVISO] El símbolo %c ya existe"*; (7) en el momento en que se detecte que es un AF no determinista se debe avisar sólo una vez *"[AVISO] Se ha detectado que el AF es no determinista"*.

Opcionalmente, se puede acabar de implementar el programa C que, unido a la función `transición` construida por BISON, permita ejecutar completamente el AF. Este programa, que será siempre el mismo, estará en un fichero `af.c` que incluirá el fichero `af.h` (`#include "af.h"`) generado por nuestro programa BISON y que contendrá la función `transición`.

ENTREGA: Fichero comprimido que contenga los siguientes documentos:

- **Una hoja** explicativa (PDF) donde aparezca el nombre y apellidos de los integrantes del grupo, el enunciado, una indicación de si se ha realizado la parte opcional y un apartado con las explicaciones que deseen hacerse.
- **El archivo af.l** (FLEX) conteniendo el programa FLEX desarrollado, una vez se hayan introducido los cambios necesarios respecto a la práctica 1.
- **El archivo af.y** (BISON) conteniendo el programa BISON desarrollado.
- **Una hoja** con un juego de pruebas de todos los errores y avisos. También se presentará un caso de generación correcta del fichero `af.h` para un AF determinista y otro para un no determinista.
- Esta entrega corresponderá a una nota entre 0 y 7 puntos de la nota de la práctica 2.
- **(opcionalmente) El archivo af.c** con el código C para ejecutar AFs, así como el programa ejecutable `af.exe`. En este caso, el juego de pruebas deberá mostrar también algunas ejecuciones de un AF determinista y de un AF no determinista. Con la entrega de este apartado, la evaluación de toda la práctica 2 será entre 0 y 10 puntos.