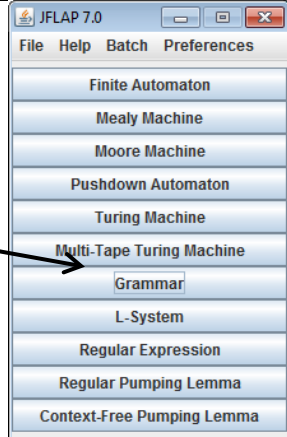


# Procesadores de Lenguajes – Analizadores sintácticos LR con JFlap

Esta pequeña práctica pretende introducir el uso de JFlap para ver el funcionamiento de analizadores LR, en concreto SLR(1). JFlap es un software de libre distribución y puede descargarse del campus virtual y también desde su sitio web <http://www.jflap.org>.

Ejecutar el fichero “JFLAP.jar”, aparecerá la siguiente pantalla, donde habrá que seleccionar la opción “Grammar”. Con esta acción aparecerá una pantalla donde especificar la gramática.



The screenshot shows the JFLAP 7.0 application window. The menu bar includes File, Help, Batch, and Preferences. The main area contains a list of options: Finite Automaton, Mealy Machine, Moore Machine, Pushdown Automaton, Turing Machine, Multi-Tape Turing Machine, Grammar (highlighted), L-System, Regular Expression, Regular Pumping Lemma, and Context-Free Pumping Lemma.

Cada fila se corresponderá con una regla de producción. LHS (Left Hand Side) es la columna para antecedentes y RHS (Right Hand Side) es la columna para consecuentes.

Los símbolos deben rellenarse sin espacios en blanco entre ellos. JFlap sólo trabaja con símbolos de una letra.

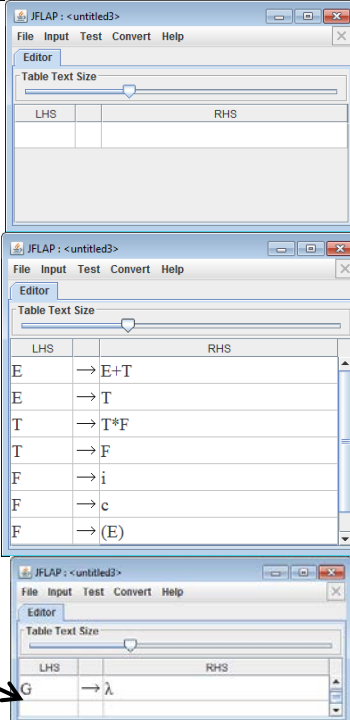
Introducir la gramática:

$E ::= E+T \mid T$

$T ::= T * F \mid F$

$F ::= i \mid c \mid (E)$

Por defecto, JFlap considera que la regla será borradora, de forma que si escribimos el antecedente y no el consecuente el resultado será una regla de este tipo



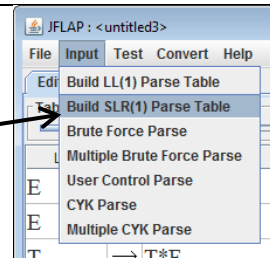
The three screenshots show the JFLAP Editor window. The first screenshot shows an empty table with LHS and RHS columns. The second screenshot shows the rules being entered: E → E+T, E → T, T → T \* F, T → F, F → i, F → c, and F → (E). The third screenshot shows the result of a rule where only the antecedent 'G' is entered, resulting in a rule G → λ.

LHS	RHS
E	→ E+T
E	→ T
T	→ T * F
T	→ F
F	→ i
F	→ c
F	→ (E)

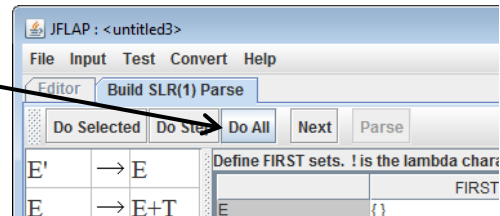
  

LHS	RHS
G	→ λ

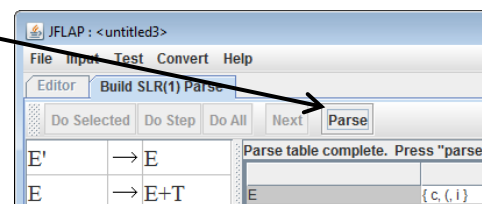
Una vez introducida la gramática en JFLAP, se debe seleccionar la opción Input > Build SLR(1) Parse Table.



En la siguiente pantalla seleccionar el botón "Do All" para construir completamente el autómata.



Una vez construido seleccionar el botón "Parse" para simular su funcionamiento.



Aparecerá la siguiente pantalla donde habrá que introducir la cadena de entrada (sin espacios en blanco) en el campo "input", seleccionar "Start", e ir ejecutando pasos de análisis con el botón "Step"

The screenshot shows the 'SLR(1) Parsing' window in JFLAP. At the top, there's a menu bar with 'File', 'Input', 'Test', 'Convert', and 'Help'. Below it, the 'Editor' tab is active, showing 'Build SLR(1) Parse' and 'SLR(1) Parsing'. The 'Table Text Size' is set to 10. The main area contains a table of LR(1) items and their transitions.

	(	)	*	+	c	i	\$	E	F	T
0	s1				s5	s6		2	3	4
1	s1				s5	s6		7	3	4
2				s8			acc			
3		r4	r4	r4			r4			
4		r2	s9	r2			r2			
5		r6	r6	r6			r6			
6		r5	r5	r5			r5			
7		s10		s8						
8	s1				s5	s6		3	11	

Below the table, there's a section for 'LHS' and 'RHS' with the following entries:

LHS	RHS
E'	→ E
E	→ E+T
E	→ T
T	→ T*F
T	→ F
F	→ i
F	→ c
F	→ (E)

At the bottom, there's an 'Input' field, 'Input Remaining', and 'Stack' sections. The 'Input' field is empty. The 'Input Remaining' and 'Stack' sections are also empty. The 'Start' and 'Step' buttons are visible. The 'Noninverted Tree' button is also present.