

UNIVERSITÀ DEGLI STUDI DI CATANIA

CORSO DI LAUREA MAGISTRALE IN INGEGNERIA INFORMATICA

**Analisi Top-Down LL(1) di grammatiche context-free**

Relazione di Linguaggi e Traduttori

**Docente: Studenti:**

Prof.ssa Vincenza Carchiolo Floriana Leone O55000246

Giuseppe Mastrosimone O55000262

Anno Accademico 2015/2016

**Introduzione**

Lo scopo di questo progetto è quello di implementare il funzionamento di un parser top-down di tipo LL(1), cioè di effettuare l’analisi sintattica di grammatiche non contestuali (context-free) andando a considerare un solo carattere di lookahead.

In particolare, l’applicazione sviluppata è in grado di verificare se una grammatica è context-free e in caso affermativo si procede con l’analisi LL(1). Il parser implementato è capace di analizzare le grammatiche inserite, verificando di volta in volta se esse soddisfino i requisiti di un parser LL(1) e in tal caso si procede alla generazione delle relative tabelle di parsing.

**Strumenti utilizzati**

L’applicativo è stato implementato utilizzando il linguaggio OO Java. In particolare, per lo sviluppo del codice sono stati usati:

* Eclipse Luna 4.4, come IDE per lo sviluppo del codice Java;
* Libreria Batik, integrata alla piattaforma di sviluppo per la realizzazione delle tabelle di parsing con grafica SVG di tipo XML-like.

**Descrizione del codice**

**Premessa sulla definizione delle grammatiche**

Al fine di effettuare l’analisi LL(1) è necessario per prima cosa definire le grammatiche alle quali è possibile applicarla. Per questo motivo l’utente ha la possibilità di inserire da tastiera i simboli terminali e non terminali e le relative produzioni in modo da definire la grammatica desiderata. In particolare, il formato della grammatica deve essere il seguente: A->aBc, dove la parte sinistra deve contenere solo un terminale (indicato con la lettera maiuscola) e quella destra può contenere invece un numero variabile di terminali e non terminali. Seguendo questo formato viene garantita che la grammatica inserita sia di tipo non contestuale, ma non si ha la certezza che sia di tipo LL(1). Terminali e non terminali sono differenziati dal fatto che solamente i primi possono contenere lettere maiuscole, mentre i secondi possono essere espressi da qualsiasi carattere purché non contenga lettere maiuscole.

**File delle grammatiche**

Una volta definito il formato delle grammatiche supportato dal programma sviluppato, è bene sapere come è possibile leggerlo o definirlo.

L’applicazione permette due modalità di definizione della grammatica di cui si vuole effettuare l’analisi LL(1): lettura della grammatica da file o scrittura della grammatica su file. Il tutto è reso possibile grazie a un menù ben definito che consente all’utente di scegliere l’opzione di sua preferenza.

In particolare, se l’utente sceglie di definire la grammatica inserendo i valori dei terminali, dei non terminali e delle produzioni da tastiera il tutto, oltre ad essere stampato a schermo, viene salvato su 2 file, ovvero: nome\_file.txt dal quale è possibile visualizzare direttamente la grammatica inserita e nome\_file (senza estensione). Questo secondo file è utile nel caso in cui l’utente decida di non fare l’inserimento da tastiera della grammatica, bensì voglia effettuare l’analisi di una grammatica già precedentemente definita in tal modo.

Inoltre, è consentita un’altra modalità di lettura da file della grammatica definita, in questo caso, esternamente dal programma stesso. Ciò vuol dire che l’utente, può anche scegliere di scrivere direttamente su un file esterno (senza usare l’applicativo stesso), seguendo le regole di definizione delle grammatiche supportate dal programma.

**Menù**

Come già anticipato, quando il programma viene mandato in esecuzione viene avviato un menù.

