

Indice

1. Prefazione	2
1.1 Version History	2
1.2 Sommario modifiche	2
2. Introduzione	3
2.1 Definizioni e glossario	3
2.2 Scopo	3
3. Requisiti funzionali	4
3.1 Struttura corretta della grammatica	4
3.2 Errori lessicali	4
3.3 Errori sintattici	5
3.4 Errori semantici	5

1. Prefazione

1.1 Version History

Versione 0.0 - 16/04/2019

1.2 Sommario modifiche

Definizione dei requisiti funzionali, dei casi d'uso, della struttura generale del software, della struttura dei test e degli strumenti utilizzati per lo sviluppo.

2. Introduzione

2.1 Definizioni e glossario

Grammatica libera da contesto

In informatica e in linguistica, una grammatica libera dal contesto è una grammatica formale in cui ogni regola sintattica è espressa sotto forma di derivazione di un simbolo a sinistra a partire da uno o più simboli a destra.

Ciò può essere espresso con due simbolismi equivalenti:

1) $S := \alpha$

2) $S \rightarrow \alpha$

dove S è un simbolo detto *non terminale*, sostituibile con altri simboli non terminali e/o con simboli terminali, e α è una sequenza di simboli non terminali e/o terminali, ossia simboli che non possono essere sostituiti con null'altro.

L'espressione "libera dal contesto" si riferisce al fatto che il simbolo non terminale S può sempre essere sostituito da α , indipendentemente dai simboli che lo precedono o lo seguono; un linguaggio formale si dice libero dal contesto se esiste una grammatica libera dal contesto che lo genera.

Parser

Un parser LR è un parser di tipo bottom-up per grammatiche libere da contesto che legge il proprio input partendo da sinistra verso destra, producendo una derivazione a destra. Laddove indicato come parser $LR(k)$, il k si riferisce al numero di simboli letti (ma non "consumati") per prendere le decisioni di parsing.

2.2 Scopo

Scopo dell'applicazione che si andrà a realizzare è il riconoscimento di una grammatica LR(1) contenuta in un file di input selezionato dall'utente.

3. Requisiti funzionali

L'applicazione deve consentire all'utente le seguenti azioni:

- selezione del file di input da sottoporre al parsing e all'identificazione.

L'applicazione deve fornire le seguenti funzionalità:

- effettuare il parsing del file ricevuto in input, individuando eventuali errori sintattici, lessicali o semantici;
- qualora vengano individuati errori di qualsiasi genere nella fase di parsing, l'applicazione deve comunicare i dettagli relativi agli errori individuati all'utente;
- qualora non vengano individuati errori nella fase di parsing, l'applicazione deve procedere nell'identificare la grammatica come grammatica $LR(1)$ o non $LR(1)$.

3.1 Struttura corretta della grammatica

L'applicazione deve riconoscere come formalmente corrette (quindi prive di errori sintattici, lessicali e/o grammaticali) soltanto grammatiche che presentino la seguente struttura:

- una prima regola pr che abbia come elemento di sinistra il non terminale S_0 , definita come segue

$$SZ \ EQ \ NT \ TER \ SC$$

- altre $n \geq 1$ regole di produzione ar , che formano il resto della grammatica, definite come segue

$$NT \ EQ \ (NT|T)^* \ SC$$

I blocchi componenti le regole appena definite sono così traducibili:

Simbolo	Caratteri
SZ	S_0
EQ	$- \ > \ \ :=$
NT	$A \ \dots \ Z$
CT	$a \ \dots \ z \ \ 0 \ \dots \ 9 \ \ + \ \ - \ \ * \ \ /$
TER	$/swa \ \ /cjswa$
SC	$;$

Tabella 1: Corrispondenza tra caratteri della grammatica e blocchi di definizione delle regole

Nota: Per la definizione della struttura delle regole è stata utilizzata la notazione formale di Backus-Naur estesa (EBNF).

3.2 Errori lessicali

L'utilizzo di qualsiasi carattere non riconducibile alla colonna "Caratteri" della Tabella 1 corrisponde a un errore lessicale.

3.3 Errori sintattici

Gli errori sintattici sono dati dal mancato rispetto della struttura delle regole *pr* e *ar* come definite nel paragrafo "Struttura corretta della grammatica" a pagina 4.

3.4 Errori semantici

Gli errori semantici si verificano nei seguenti casi:

- nella grammatica è presente un carattere non terminale che non presenta regole di produzioni associate;
- nella grammatica è presente una regola duplicata (nota bene questo non è un errore bloccante).