

Generali

Come si specifica il lessico di un linguaggio?

Il lessico descrive le parole o elementi lessicali che compongono le frasi; la specifica del lessico coinvolge la definizione dei token che compongono il linguaggio.

Come si specifica la sintassi di un linguaggio?

La specifica della sintassi coinvolge la definizione della struttura grammaticale del linguaggio. Si tratta di stabilire le regole che determinano come combinare i token del lessico per formare costrutti validi nel linguaggio. La sintassi è definita tramite grammatiche formali.

Cosa si fa durante l'analisi lessicale?

Durante l'analisi lessicale si verifica se le sottostringhe del testo sorgente corrispondono a elementi lessicali validi, e vengono tradotte in un'opportuna codifica che faciliti la successiva elaborazione da parte del traduttore o interprete. Gli elementi lessicali nei quali la frase viene suddivisa sono detti token. Inoltre vengono eliminati spazi bianchi e commenti.

Cosa si fa durante l'analisi sintattica?

Durante l'analisi sintattica vengono analizzati i token per verificare che la loro struttura sia conforme alle regole della grammatica. Inoltre, viene costruita una rappresentazione interna della struttura dei token analizzati, l'AST.

Cosa si fa durante l'analisi semantica?

Durante l'analisi semantica si controlla che le variabili siano correttamente dichiarate e che i tipi siano corretti. Queste informazioni non possono essere rappresentate con una CFG. In questa fase, che dipende dalle regole semantiche del linguaggio sorgente, il type-checker decora l'AST generato dal parser aggiungendoci informazioni di tipo.

Quali sono gli input e gli output delle 3 analisi precedenti?

1. Input: programma sorgente, Output: tokens.
2. Input: tokens, Output: AST (e symbol table).
3. Input: AST, Output: AST decorato (informazioni semantiche sul programma).

Scanner

Quali classi lessicali fanno in generale parte di un linguaggio di programmazione?

- *Parole chiave*: sono particolari parole fisse che caratterizzano vari tipi di frasi o strutture; ad esempio `if`, `for`, `class`.
- *Delimitatori e operatori*: come i precedenti, sono delle parole fisse composte di caratteri anche non alfabetici.
- *Classi lessicali aperte*: queste comprendono un numero illimitato di elementi lessicali, che devono avere la struttura del linguaggio regolare, ossia a stati finiti. Esempi tipici sono:

- *nomi o identificatori* di variabili, di funzioni, o altro
- *costanti*, quali numeri interi, reali o stringhe alfanumeriche
- *Commenti*: fanno parte del lessico, ma sono ignorati dalle fasi successive dell'analisi lessicale.

Cosa è un token?

Un token descrive un insieme di caratteri che hanno lo stesso significato, e sono i simboli usati per definire la sintassi del linguaggio.

Come (con quale classe di linguaggi) si specificano i token dei linguaggi?

I token sono descritti da espressioni regolari e riconosciuti da automi a stati finiti.

Come di può implementare il riconoscimento lessicale?

Si può realizzare:

- proceduralmente, a partire da espressioni regolari, automi a stati finiti e grammatica regolare
- tramite tabella interpretata; una struttura dati rappresenta il DFA riconoscitore della grammatica e un programma realizza il funzionamento della DFA
- automaticamente, con un generatore di scanner

Parser

Definizione di grammatica LL(1)

Una grammatica è LL(1) è una grammatica context free analizzabile da un parser LL(1). Una grammatica è LL(1) se, per ogni simbolo non terminale, un token predice al più una produzione.

Per quali ragioni una grammatica può non essere LL(1)?

Una grammatica ricorsiva sinistra non può essere LL(1) in quanto può causare loop infiniti e generare ambiguità. Ad esempio, considerando la regola $A \rightarrow A\alpha|\beta$:

- se $a \in FIRST(\beta)$, allora $a \in FIRST(A\alpha)$, perché $A \Rightarrow A\alpha \Rightarrow \beta\alpha$
- se $\beta \rightarrow \varepsilon$, allora $a \in FOLLOW(A)$ se $a \in FIRST(\alpha)$, e quindi anche $a \in FIRST(A\alpha)$, perché $A \Rightarrow A\alpha \Rightarrow \alpha$

Una grammatica con prefissi comuni non può essere LL(1).

Come si può specificare un parser Top-Down?

AST e Symbol Table

Cosa è e a cosa serve definire un Abstract Syntax Tree per una grammatica di un linguaggio di programmazione?

Quale è la differenza fra AST e parsing tree per una stringa di un linguaggio di programmazione?

Cosa è la Symbol Table, quali informazioni contiene e a cosa serve?

Come può essere implementata una Symbol Table?

Analisi di Tipo e Visitor

Cosa si fa con l'Analisi di Tipo? Come si specifica?

A cosa serve il Pattern Visitor? Descriverne la struttura.