Context Free Grammar

Module 3

Definition of a Context Free Grammar

Definition 4.1 Represented by 4-tuple (V_N,T, P , S)where

(i) V is a set of variables

(ii) ∑ or T is a set of terminals

(iii) S is a start symbol and S ∈ V

(v) P is a set of productions or rules of the form A->α where A is a variable and α is a string of variables and terminals i.e. α ∈ (V U ∑)*

Context free Grammar

- Have a natural recursive notation
- Defines Parse trees
- Parse tree is a picture of the structure that a grammar places on the strings of its language.
- Pushdown Automata describes all and only the context free languages

Applications of Context free Grammar

- Parse tree construction and Compiler design

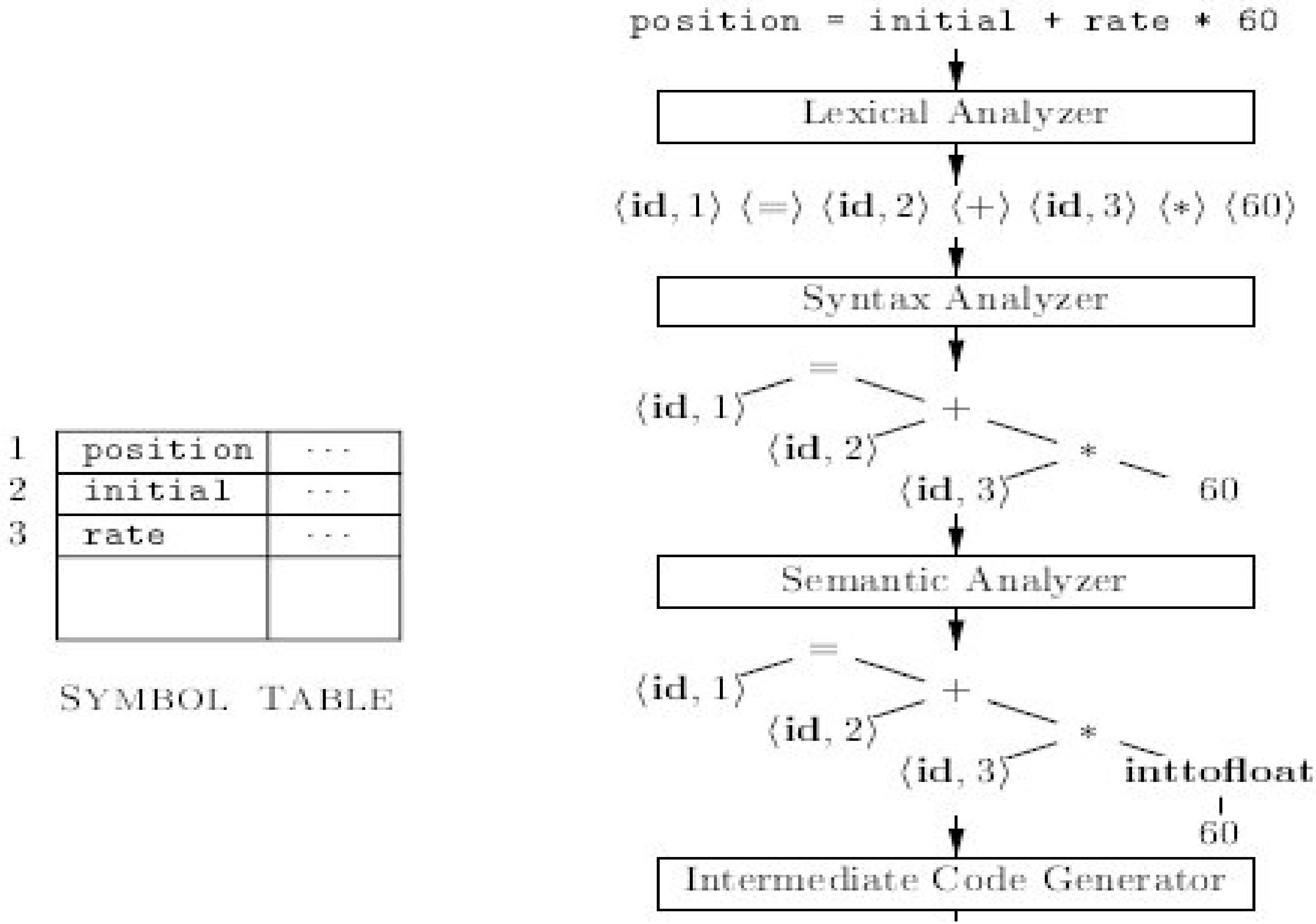
- XML, Document type Definition (DTD)

Applications of Context free Grammar

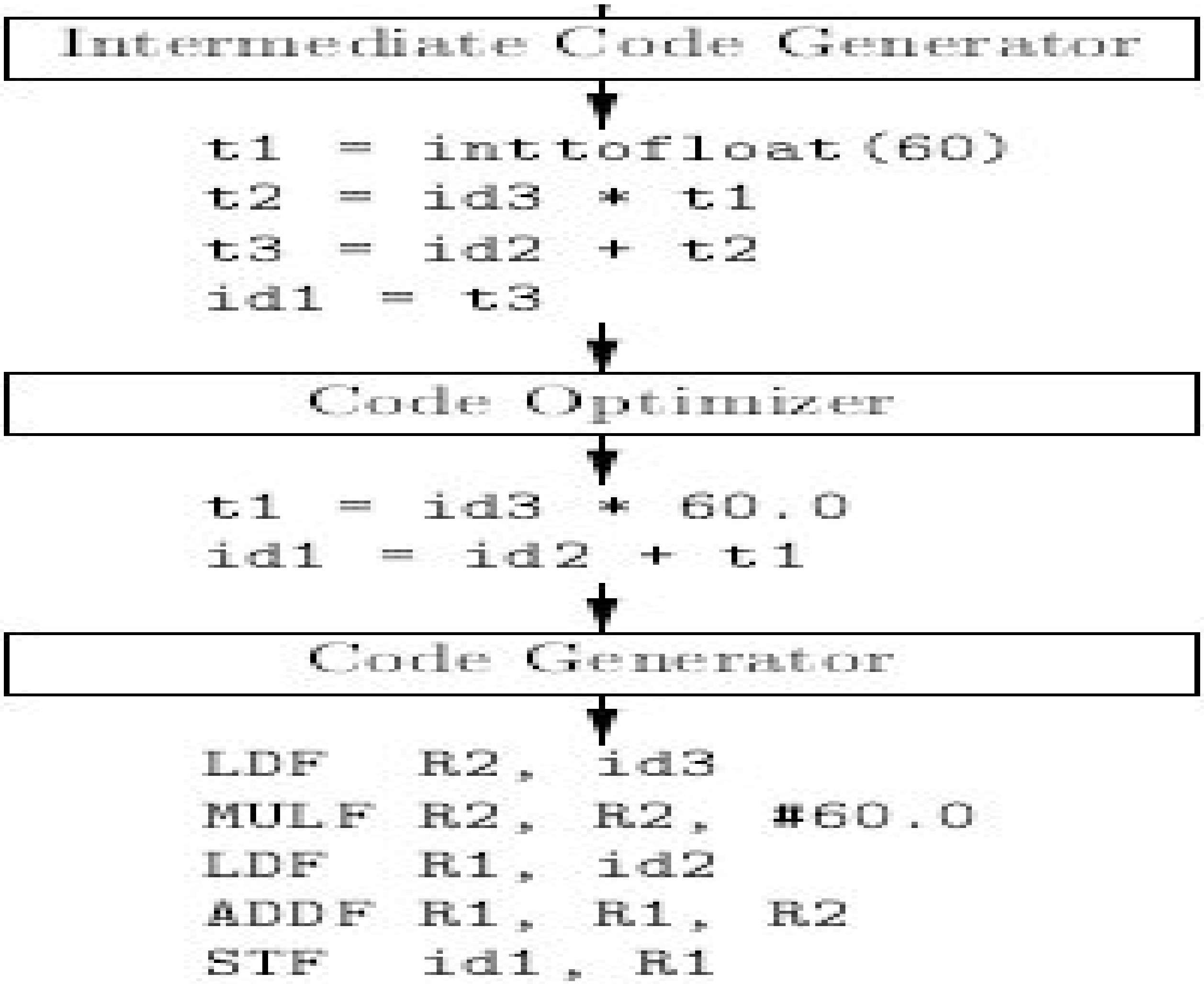
- Played a central role in Compiler technology since the 1960s

- They turned the implementation of parsers from a time consuming, ad-hoc implementation task into a routine job that could be done quickly

Example Revisited



Example Revisited



Context free Grammar:Example 1

G=({S,C},{a,b},P,S} where P consists of

S->aCa

C->aCa|b

Find the language generated by G

Context free Grammar:Example 1

G=({S,C},{a,b},P,S} where P consists of

S->aCa

C->aCa|b

Find the language generated by G

S->aCa	by using S->aCa
->aaCaa	by using C->aCa
->aaaCaaa	by using C->aCa
->aaaabaaa	by using C->b

$L(G)=\{a^n b a^n | n>=1\}$

Context free Grammar:Example 2

G=({S},{a,b},{S->aSb, S->ab},S}

Find the language generated by G

Context free Grammar:Example 2

G=({S},{a,b},{S->aSb, S->ab},S}

Find the language generated by G

S->aSb	by using S->aSb
->aaSbb	by using S->aSb
->aaaSbbb	by using S->aSb
->aaaabbbb	by using S->ab

$L(G)=\{a^n b^n | n>=1\}$

Context free Grammar:Example 3

Construct CFG to generate a set of palindromes over the alphabet {a,b}

Context free Grammar:Example 3

Construct CFG to generate a set of palindromes over the alphabet {a,b}

When we read palindrome in forward or reverse manner, we get the same thing. In a Palindrome the start and end symbols are same.

Thus, the productions are

S->aSa|bSb

S-> ε|a|b

G=({S},{a , b},{ S->aSa|bSb, S->ε|a|b },S}

Context free Grammar:Example 4

Construct a CFG generating $L(G) = \{0^n 1^{2n}, n>=1\}$

Context free Grammar:Example 4

Construct a CFG generating $L(G) = \{0^n1^{2n}, n>=1\}$

Properties of L:

Start with 0 and end with 1

Number of 1's is two times greater than 0

Thus, the productions are

S->011

S->0S11

G=({S},{0,1},{ S->011|0S11 }, S}

Context free Grammar:Example 5

Consider the alphabet $\sum$ ={a,b,(,),+,*,., ε }. Construct a CFG that generates all strings in $\sum^*$ that are RE over the alphabet {a , b}

Context free Grammar:Example 5

Consider the alphabet ∑ ={a,b,(,),+,*,., ε }. Construct a CFG that generates all strings in ∑ * that are RE over the alphabet {a , b}

L=a*,b*,a + b, a . b, (a + b) ,a , b, ε

R->R+R
R->R.R
R->(R)
R->R*
R->a|b| ε

G=({R}, {a,b,(,),+,*,., ε },{R->R+R|R.R|(R)|R*|a|b| ε} , R}

Example

Construct a CFG generating $L(G)=\{ a^i b^j c^k \mid i, j, k \geq 0 \text{ and } i+j = k \}$

Example

Construct a CFG generating $L(G)=\{ a^i b^j c^k \mid i, j, k \geq 0 \text{ and } i+j = k \}$

- $S \rightarrow aSc \mid X$
- $X \rightarrow bXc \mid \varepsilon$

Lecture 29/3/2022

Derivation

Paramasivan-

- The process of deriving a string from start symbol by applying the production of CFG

Hopcraft-

- Use the productions from Head to Body
- Expand the start symbol using one of its productions
- Expand the resulting string by replacing one of the variables by the body of one of its productions
- Until we drive a string consisting entirely of terminals

Derivation-Example 7

Consider the grammar

G=({E},{+,*,(,),id },{E->E+E|E*E|(E)|id},E). Derive the string (id+id)*id

Derivation-Example 7

Consider the grammar
G=({E},{+,*,(,),id },{E->E+E|E*E|(E)|id},E). Derive the string (id+id)*id

E=>E*E	By using E->E*E
=>(E)*E	By using E->(E)
=>(E+E)*E	By using E->E+E
=>(id+E)*E	By using E->id
=>(id+id)*E	By using E->id
=>(id+id)*id	By using E->id

$E \overset{*}{\Rightarrow} (id+id)*id$

Derivation-Example 8

Consider the grammar

G=({E},{+,*,(,),id },{E->E+E|E*E|(E)|id},E). Derive the string id+(id*id)

Derivation-Example 8

Consider the grammar

G=({E},{+,*,(,),id },{E->E+E|E*E|(E)|id},E). Derive the string id+(id*id)

E=>E+E	By using E->E+E
=>E+(E)	By using E->(E)
=> E+(E*E)	By using E->E*E
=> id+(E*E)	By using E->id
=> id+(id*E)	By using E->id
=> id+(id*id)	By using E->id

$$E \overset{*}{\Rightarrow} id+(id*id)$$

Types of Derivation

- Left Most Derivation(LMD)
- Right Most derivation(RMD)

Types of Derivation

Left Most Derivation(LMD)-

- A derivation $A \overset{*}{\Rightarrow} w$ is called Left most derivation if we apply a production only to the left most variable at every step

Right Most derivation(RMD)

- A derivation $A \overset{*}{\Rightarrow} w$ is called Right most derivation if we apply a production only to the right most variable at every step

-KLP Mishra

Context free Grammar:Example 9

Consider the grammar S->aAS, A->SbA, S->a,A->ba Derive the string aabbaa using LMD

Context free Grammar:Example 10

Consider the grammar S->aAS, A->SbA, S->a,A->ba Derive the string aabbaa using RMD

Context free Grammar:Example 11

Consider the grammar G=({S},{a,b},

{ S->0B|1A,

A->0|0S|1AA,

B->1|1S|0BB}, S}

For the string 0110, find RMD.

Context free Grammar:Example 12

Consider the grammar G=({S},{a,b},{

S->0B|1A,

A->0|0S|1AA,

B->1|1S|0BB}, S}

For the string 00110101, find the following:

a) LMD

Context free Grammar:Example 12

Consider the grammar G=({S},{a,b},{

S->0B|1A,

A->0|0S|1AA,

B->1|1S|0BB}, S}

For the string 00110101, find the following:

b) RMD

Definition of a Derivation tree

- A tree representation of derivations is called derivation trees or parse trees
- It is useful in compilation of programming languages
- The tree structure of the source program makes the translation of source program into executable code easier

Definition of a Derivation tree

A derivation tree/Parse tree for a CFG G (V_N, ∑, P , S) is a tree satisfying the following:-

1) Every vertex/node of the tree has a label which is variable or terminal or ε
2) The root has label S
3) The label of an internal node is a Variable
4) If the interior node is labelled with A and the sons of the vertex are labelled with $X_1X_2....X_n$ from the left then $A \rightarrow X_1X_2....X_n$ must be a production
5) A node is a leaf if its label a∈ ∑
6) If an vertex has label ε then it is a leaf and it is the only one son of its father

Yield of a Derivation tree

Definition:-

Parmasivan-

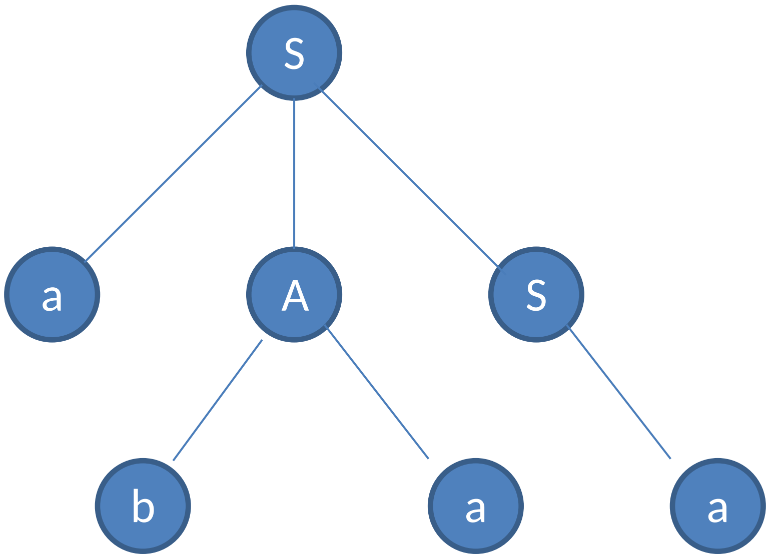
- If we read the leaves of the derivation tree from left to right order , we get the string or the expression. This is called yield of the tree

KLP Mishra-

- The yield of a derivation tree is the concatenation of the labels of the leaves without repetition in the left to right ordering

Yield of a Derivation tree

Example-Yield of the tree ?



Context free Grammar:Example 14

Find a derivation tree of a*b+a*b where G is given by

S->S+S|S*S|a|b

Context free Grammar:Example 14

Find a derivation tree of a*b+a*b where G is given by

S->S+S|S*S|a|b

Context free Grammar:Example 13

Consider the grammar G=({S},{a,b},{ S->aAS|a, A->SbA|ba}, S}

Show that $S \overset{*}{\Rightarrow} aabbaa$ and Construct a derivation tree whose yield is aabbaa

Sentential Form

- If G=(V_N, ∑, P , S) is a CFG then any string α in $(VUT)^*$

  such that $S \overset{}{\underset{*}{=>}} \alpha$ is a sentential form

- $S \underset{lm}{\overset{}{\underset{*}{=>}}} \alpha$, then α is left sentential form
- $S \underset{rm}{\overset{}{\underset{*}{=>}}} \alpha$, then α is right sentential form

Ambiguity in Grammar

- The following sentence in English Language:-

"In books selected information is given"

- This Sentence may be parsed in two different ways

- Same situation may arise in CFG

Ambiguity in Grammar

- A CFG is said to be ambiguous if there exist some $w \in L(G)$ that has atleast two or more Leftmost Derivation trees or Rightmost Derivation trees.

Ambiguity in Grammar :Example 15

Consider the grammar G=({S},{a,b,+,*},P,S) where P consists of S->S+S|S*S|a|b. Check whether G is ambiguous for the string a+a*b

Ambiguity in Grammar :Example 15

Consider the grammar G=({S},{a,b,+,*},P,S) where P consists of S->S+S|S*S|a|b. Check whether G is ambiguous for the string a+a*b

1st Left Most Derivation	2nd Left Most Derivation
S=>S+S	S=>S*S
=>a+S	=>S+S*S
=>a+S*S	=>a+S*S
=>a+a*S	=>a+a*S
=>a+a*b	=>a+a*b

Two LMDS , Thus Ambiguous

Ambiguity in Grammar :Example 16

- Consider the grammar $S \rightarrow aS|aSbS|\varepsilon$. Show that G is ambiguous for the string aab. Further draw their derivation trees

Ambiguity in Grammar :Example 17

- If G is the grammar S->SbS|a , Show that G is ambiguous .

Ambiguity in Grammar :Example 17

- If G is the grammar S->SbS|a , Show that G is ambiguous .
- Lets take string abababa