Lecture 6

Context Free Language (CFL) & Context Free Grammar (CFG)

**Context Free Language (CFL)**

* These Language are consists of language free.
* What does context free mean ?
* Context-sensitive grammars allow more than one symbol.
* Like : xAy → x(S)y can only be applied to the non-terminal A when it is in the *context* of x and y
* It means all of its production rules have a single non-terminal on their left hand side.
* For example, this grammar which recognizes strings of matched parentheses ("()", "()()", "(())()", ...) is context-free:

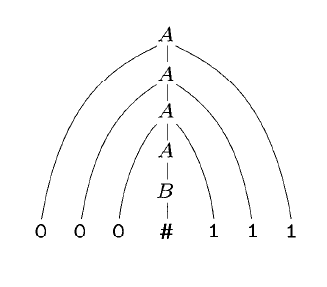
S SS

S (S)

S ()

* The left-hand side of every rule consists of a single non-terminal (in this case it's always  S but there could be more.)
* Language Recognizer :
* Think yourself as a language processor
* Recognizer can accept valid strings
* Finite automata, Regular expression are formalized type of recognizer.
* Language Generation:
* Consists of language that have recursive structure
* Context Free Grammar (CFG) is the method of generating class of context free language using some rule
* Used mostly in compilers and interpreter to design parser

**Context Free Grammar**

* Grammar G1:
  + A 0A1
  + A B
  + B #
* A collection substitution rule called production
* Variable / non Terminal
* Terminal
* Start Variable
* Terminal:
* A terminal is a symbol which does not appear on the left-hand side of any production.
* A grammar contains a set of terminal symbols (tokens) such as the plus sign, +, the times sign, \*, and other tokens defined by the lexical analyzer such as *Identifiers*
* Non-Terminal:
* Nonterminal are the non-leaf nodes in a parse tree. In the Expression grammar, E, T, and F are nonterminal.
* Sometimes nonterminal are enclosed between angle brackets to distinguish them from terminals.
* Grammar G1 Generates the string 000#111 .
* The sequence of substitution to obtain a string is called deviation
* The deviation for that string is :
* A0A100A11000A111000B111000#11
* Parse tree of Grammar G1:

**Context Free Language**

* Definition of CFL:
* A collection of languages associated with CFG is called CFL
* L(G1) language generated by grammar G1
* L(G1) is {0n # 1n | n 0}
* In shortly :
* Any language generated by CFG is called CFL

**Context Free Grammar (cont.)**

* Prove using the regular expression R=a(a\*U b\*)b that CFG is a language Generator
* CFG is Language Generator:
  + R=a(a\* U b\*)b
  + Let M =(a\* U b\*)
  + So, S= a M b
  + M= A | B
  + A= | aA
  + B = | bB

Now , we need to remember that

* + a\* = | aA
  + a+ = aA
  + b\* = | aB
  + b+ = bB

So, Sample derivation for the string S= a M b will be….

aMbaAbaaAbaaaAbaaaaaab

So, this string is accepted by this grammar

* Formal Definition of CFG
* A CFG is a 4 tuples G=(*V, , R, S*) where,

1. *V* is a finite set called the variables
2. is a finite set , disjoint from V, called terminals
3. *R* is a finite set of rules, with each rule begin a variable and a string of variables and terminals
4. *S V* is a start variable

* For, Grammar G ,
  1. A 0A1
  2. A B
  3. B#
* From the formal definition we can write,

V = {A, B}

= {0,1}

S = A

R= { A 0A1, AB, B# }

* Simple Grammar G3:
* |
* |
* |
* |
* *boy* | *girl* | *flower*
* *touches* | *likes* | *sees*
* *with* | *to* | *at*

In Grammar G3

* No of Rules => 18
* No of non Terminals / Variables => 10
* No of Terminals => 26 + 1 (space character)

Strings in L(G3) include:

* a boy sees
* the boy see a flower
* a girl with a flower likes the boy

**String** **Derivation**

* The derivation

a

a

a

a

a

* Self Study
* Derivation of other two strings