Cmp313 summary

Properties of syntax

* Root is labelled by start symbol
* Each leave is a terminal or null symbol

An alphabet of a vocabulary is a finite set of symbols or characters. (0,1) are alphabet of computer.

A string is a finite sequence of letters from an alphabet.

A sentence is a string of (non) terminal symbols derived from start symbols which follow one or more production rule.

Language is a set of sentences derivable using a given grammer

* Binary string is a string drawn from alphabet 1,0
* Kleene star (∑\*) denotes the set of all finite strings on the alphabet ∑
* ∑∞ denotes the set of al infinite string from the set of ∑ including epsilon (∈)
* Grammar can be described as a 4 tuple: G = (T, N, P, S)
* Syntax parse tree: a tree diagram that describes the syntax by breaking it down to its constituent parts of the sentence
* Parsing: is a process of recognizing a string in a language by breaking it down and analyzing each symbol against the grammar. The tree generated from parsing process is a parse tree.
* Kleene closure ∑+ denote s one or more occurrences of the preceding elements

Properties of grammar

* Equivalence of grammar: when two parse trees generate the same language, they are equivalent. Although they may be equivalent, they may not generate the same parse tree.
* Ambiguity of grammar: when a grammar permits more than one parse tree from the same sentence, it is said to be ambiguous.

Example: S -> AA, A -> x/xx

TYPES OF COMPUTER GRAMMAR

Note that all type 3 are type 2, all type 2 are type 1 and all type 1 are type 0 meaning 3 -> 2 -> 1 -> 0

* TYPE 0: also called free grammar is the most general grammar.it is identified by the turning machine there are no restrictions at the left and right hand side => bCe -> bvd
* TYPE 1: also called context sensitive or dependent grammar. LHS should not contain more than one non- terminal symbol. It is recognized by linear binding machine. => Cb -> Ss
* TYPE 2: LHS consists of only a single non-terminal symbol. RHS may contain one or more non-terminal symbols. They are called context free grammar and the machine that reads this grammar are the push down machine/ automaton.

Examples are: S -> Ab, A -> a, Q -> ∈

* TYPE 3: LHS has only one non-terminal symbol and RHS contains only a single terminal symbol or an empty string. It is also known as regular grammar. The machine that can read this is called a finite state machine/ automaton. => W -> e
* The sentential form is a string of symbols derived from start symbols which contains non terminal and terminal symbols. The production rule of an input scanned and replaced from left to right is called left most derivation (LMD). The sentential form off LMD is called left sentential form, while the production rule derived from right to left is call right most derivation (RMD). the sentential form derived from RMD is called Right sentential form. It is a sequence of production rules in order to get the input string while parsing.
* These are the conditions for production rule decisions
* Production rule to be implemented
* Non-terminal symbol to be replaced
* Recursion: production that are defined in terms of themselves
* Left factoring: production routes with identical parts to alternative right hand side
* to generate a string from a language, you need: Start, Production, Nonterminal and Terminal

Concatenation is defined as the sequence of characters of a set y followed by a sequence of characters from another set x denoted by yx

Types of string length

* Fixed length string: all have a fixed maximum length whether it has been reached or not
* Variable length string: length depends on the strings actual size.

Former Grammar: it is a set of rules for forming or rewriting strings in a formal language

Parsing: is a process of recognizing an utterance by breaking it down into a set of symbols and analyzing each against the grammar

Methods of parsing

* Start with the sentence symbol and build down towards the terminal symbols (top-down). From root to leave
* Start from the terminal symbols and build up to the root symbols (bottom up)

Properties of a parse tree

* The root is labelled by a start symbol
* Each leave is labeled by a terminal or null symbol
* Each interior node is labelled by a nonterminal symbol
* It consists of one or more nodes
* There exist one or more production rules for nonterminal symbol labelling internal and external node