PUSH DOWN AUTOMATA

Prof.PRIYANGA K.K, Computer Science Department

NON-DETERMINISTIC PUSH DOWN AUTOMATA

A pushdown automaton consists of 7-mple, namely (Q, ∑ , Γ ,∂, qo, Zo, F).

- **(i) a finite nonempty set of states denoted by Q,**
- **(ii) a finite nonempty set of input symbols denoted by ∑,**
- **(iii) a finite nonempty set of pushdown symbols denoted by Γ,**
- **(iv) a special state called the initial state denoted by qo**
- **(v) a special pushdown symbol called the initial symbol on the pushdown store denoted by Zo.**
- **(vi) a set of final states, a subset of Q denoted by F, and**
- **(vii) a transition function ∂ from Q x (∑u {^}) x Γ to the set of finite subsets of Q x Γ*.**

DETERMINISTIC PUSH DOWN AUTOMATA

A pda A = (Q, ∑ , Γ ,∂, qo, Zo, F).is deterministic if (i) ∂ (q,a, Z) is either empty or a singleton. and (ii) ∂ (q,^,Z) ≠Ø implies ∂ (q, a, Z) = Ø for each a ϵ ∑

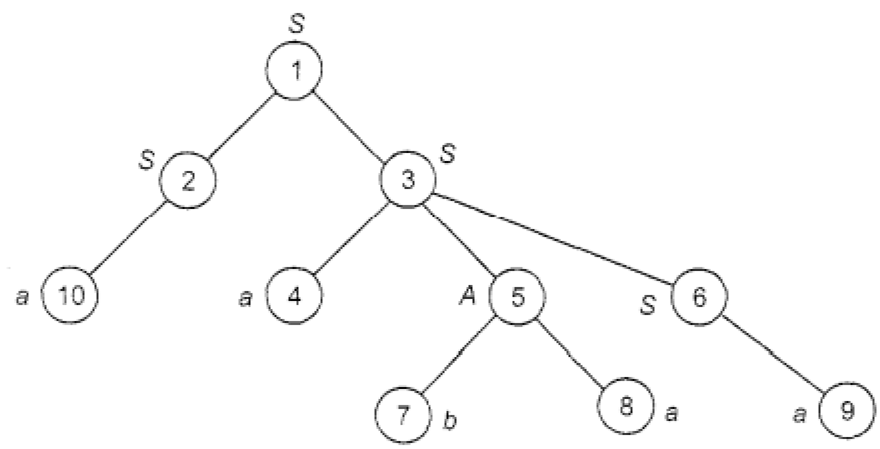
DERIVATION TREES

A derivation tree (also called a parse tree) for a CFG G = (Vn,∑, P, S) is a tree satisfying the following conditions:

- **(i) Every vertex has a label which is a variable or terminal or ^.**
- **(ii) The root has label S.**
- **(iii) The label of an internal vertex is a variable.**
- **(iv) If the vertices n1,n2,n3….,nk written with labels X1, X2, ..., Xk are the sons of vertex n with label A, then A -> X1, X2, ..., Xk is a production in P.**
- **(v) A vertex n is a leaf if its label is a ϵ ∑ or ^; n is the only son of its father if its label is ^**

YIELD OF A DERIVATION TREE

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



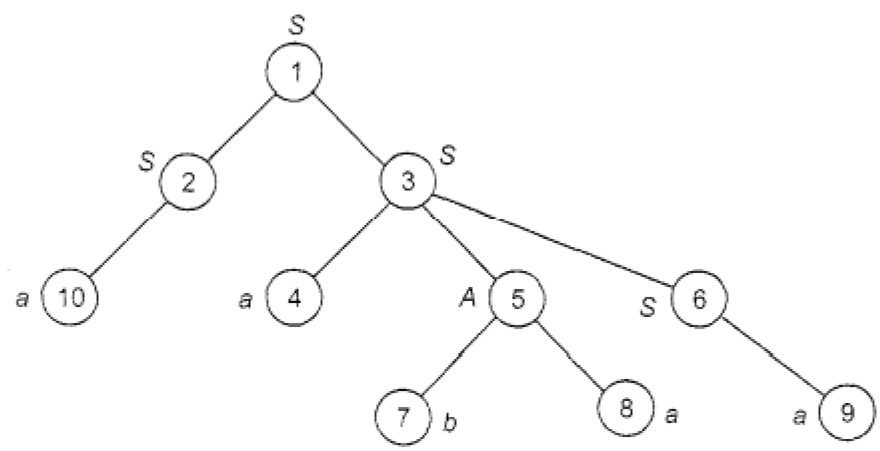
The yield of the derivation tree of is aabaa.

SUB TREE

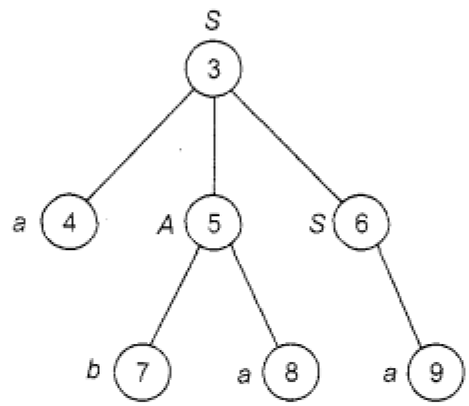
A subtree of a derivation tree T is a tree

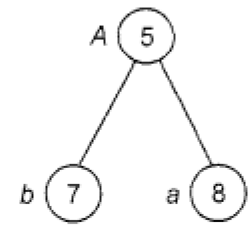
(i) whose root is some vertex v of T.
(ii) whose vertices are the descendants of v together with their labels, and
(iii) whose edges are those connecting the descendants of v.

Eg:



The two subtrees of the derivation tree given above are





<u>NOTES:</u>

A subtree looks like a derivation tree except that the label of the root may not be S. It is called an A-tree if the label of its root is A

LEFT MOST DERIVATION

A derivation A =* > w is called a leftmost derivation if we apply a production only to the leftmost variable at every step.

RIGHTMOST DERIVATION

A derivation A =* > w is a rightmost derivation if we apply production to the rightmost variable at every step

AMBIGUITY IN CONTEXT-FREE GRAMMARS

A terminal string W ϵ L(G) is ambiguous if there exist two or more derivation trees for w (or there exist two or more leftmost derivations of w)

SIMPLIFICATION OF CONTEXT-FREE GRAMMARS

1. **ELIMINATION OF NULL PRODUCTIONS**
   A production of form A -> Λ, where A is a variable is called Null production
2. **ELIMINATION OF UNIT PRODUCTIONS**

   A productions of the form A → B, A,B ϵ Vn is called Unit production

3. **REMOVAL OF USELESS PRODUCTIONS**

   A production does not derive any terminal string is called Useless productions

NORMAL FORMS FOR CONTEXT-FREE GRAMMARS

1. **CHOMSKY NORMAL FORM**

A context-free grammar G is in Chomsky normal form if every production is of the form A ->α, or A -> B C, and S ->Λ is in G if Λ ϵ L(G).

2. GREIBACH NORMAL FORM

A context-free grammar is in Greibach normal form if every production is of the form A ->aα. where α ϵ Vn* and a ϵ ∑ (α may be A), and S -> Λ is in G if Λ ϵ L(G).