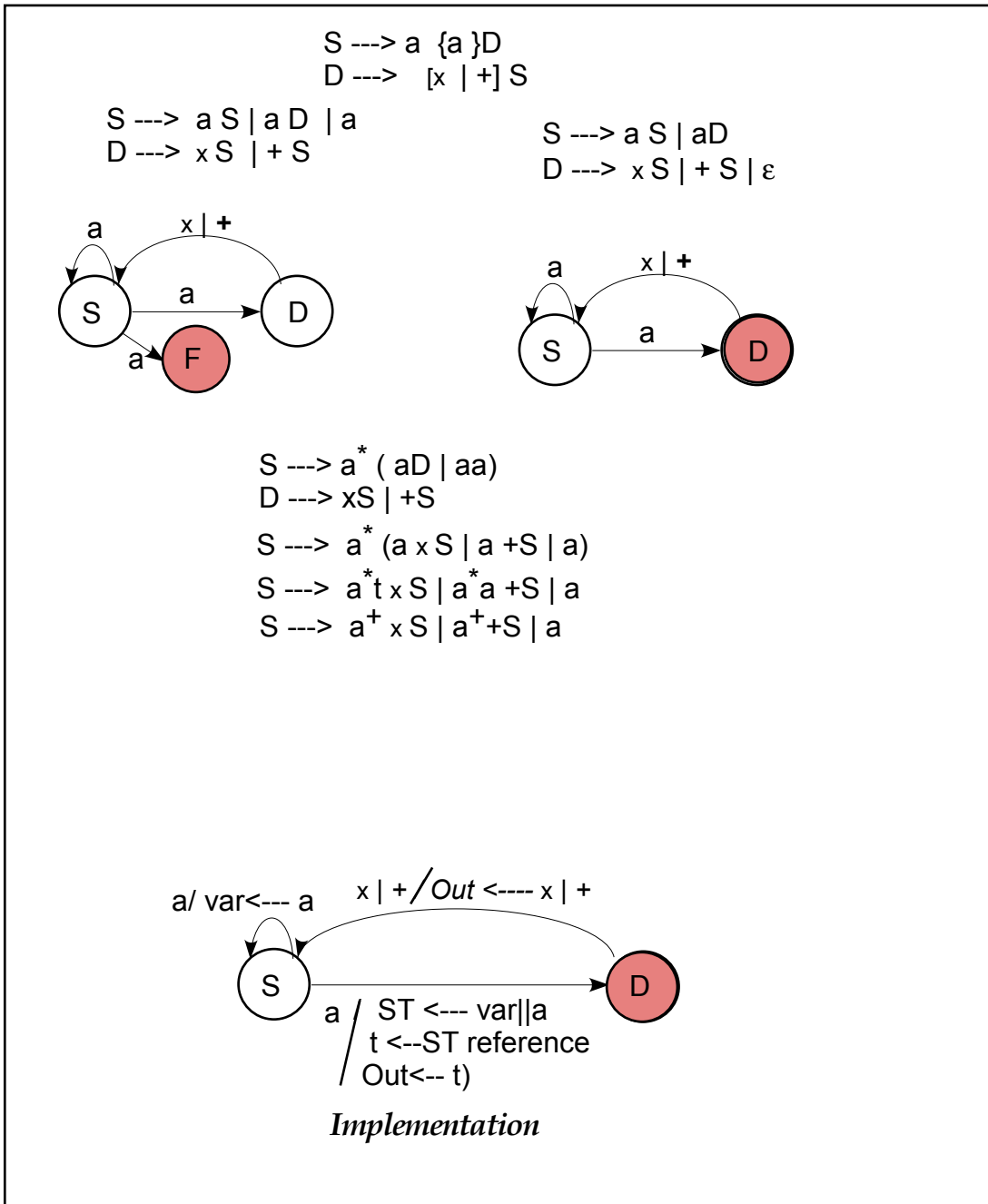


LEX -----> PARSER

Typical Input String: aa + aaa xaa + a +aa xaaaa

The language to be dealt with can be described with the following grammar(s) G-Usefull in LEX



LEX

Typical Input String: $tt + tttxtt + t + ttxtttt$

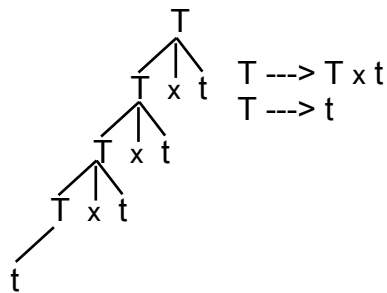
Now in the second Unit, the Parser, a grammar that will produce a bottom up Parse

$T \rightarrow T x t$

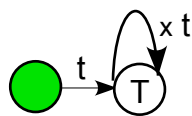
$T \rightarrow t$

so $L(T)$ is $t(xT)^*$ = t followed by any number of occurrences of $x t$

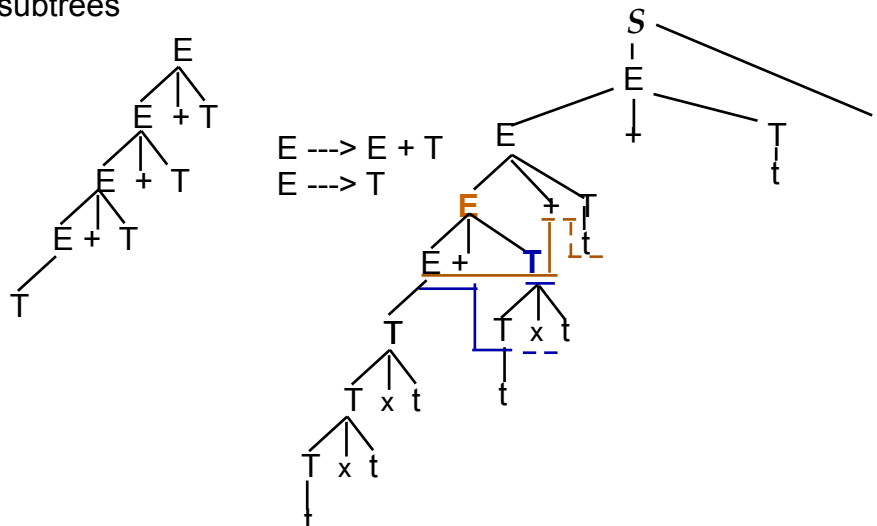
A Parse Tree looks like (a) and so could be parsed bottom up.



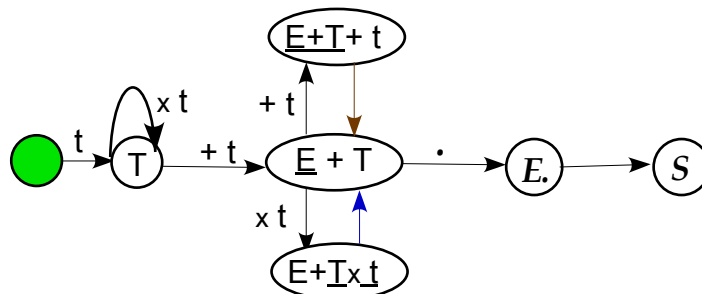
Parsing Bottom up here is done simply as follows



Consider a language with two levels of precedence. The lower precedence level is $+$ the higher is x . To parse this bottom up we need develop a grammar in which $+$ will be in upper level subtrees and which will produce parse trees in which x will be at the lower subtrees



Moving along the bottom the first that will be found is t and that must come from $T \rightarrow t$. As long as $x t$ is found the parse continues simply as shown. When $+$ appears one must change T to E and look further in deciding what to do.



- $S \rightarrow E.$
- $E \rightarrow E + T$
- $E \rightarrow T$
- $T \rightarrow T x t$
- $T \rightarrow t$

PARSER

Proofs:

headlist (L) is the list of all members excluding the last

Recursive Definition

headlist(L) = () if cdr(L) == []

headlist(L) = cons(car(L), headlist(cdr(L))) if cdr(L) != []

We claim with L = [x1, x2, ..., xn] representing any list of n elements

headlist([x1]) = [] if L = [x1]

headlist([x1, x2, ..., xn]) = [x1, ..., xn-1] if L != [x1]

Proof requires the definition equalities to hold under the assumed solution

Equalities Hold if L = [y1, y2, ..., yn] and car [y1, y2, ..., yn] = y1, cdr[y1, y2, ..., yn] = [y2, ..., yn]

and cons(y [z1, ..., zn]) = cons[y, z1, ..., zn] - yj, zj are any atoms or lists which are members of L

Proof

headlist([x1, x2, ..., xn]) =? [] if cdr([x1, x2, ..., xn]) == [] , i.e., if L =[x1]

headlist([x1, x2, ..., xn]) =? cons(car([x1, x2, ..., xn]), headlist(cdr([x1, x2, ..., xn])) if cdr([x1, x2, ..., xn]) != []
=? cons(x1, headlist([x2, ..., xn])) if cdr([x2, ..., xn]) != []
=? cons(x1, [x2, ..., xn-1]) if cdr([x2, ..., xn]) != []
== [x1, x2, ..., xn-1]

member(x, L) = #T if x is on of the elements of list L

Recursive Definition

member?(x, L) is #T if x is a member of list L

member?(x, L) = #F if L = []

member?(x, L) = #T if L != [] & equal?(x, car(L)) & L != []

member?(x, L) = member?(x, cdr(L)) if L != [] & !equal?(x, car(L)) & L != []

We claim with L = [x1, x2, ..., xn] representing any list of n elements and following solution then equalities hold

member?(x, [x1, x2, ..., xn]) = #T if x = xj for some j

member?(x, [x1, x2, ..., xn]) = #F if x != xj for any j

Given the definition cdr and car above, and equal?(x,y) being true iff x == y we do the proof by establishing the equalities in the definition under different conditions with the assumed meaning of member

member?(x, []) = #F if L = []

member?(x, [x1, x2, ..., xn]) = #T if L != [] & equal?(x, car(L)) & L != []

member?(x, [x1, x2, ..., xn]) = member?(x, cdr([x1, x2, ..., xn])) if L != [] & !equal?(x, car(L)) &

```
Mem = #F
E = pt to x1
while(Next != [])
{ if( c(E) = x) ( Mem = #T), break
  E=NEXT(E);
}
```

Proofs

$\{(X + Y + Z)/3 \leftarrow \text{average}(X+Y, Y, Z)\}??$

$X \leftarrow X + Y$

$\{(X + Z)/3 \leftarrow \text{average}(X, Y, Z)\}$

$W \leftarrow X + Z$

$\{W/3 \leftarrow \text{average}(X, Y, Z)\}$

$A \leftarrow W/3$

$\{A \leftarrow \text{average}(X, Y, Z)\}$

$\{(X + Y + Z)/3 \leftarrow \text{average}(X, Y, Z)\}$ True by definition

$W \leftarrow X + Y$

$\{(W + Z)/3 \leftarrow \text{average}(X, Y, Z)\}$

$W \leftarrow W + Z$

$\{W/3 \leftarrow \text{average}(X, Y, Z)\}$

$A \leftarrow W/3$

$\{A \leftarrow \text{average}(X, Y, Z)\}$

$A > B \implies \{B = \text{median}(A, B, 2A)\} \ \& \ A \leq B \implies \{B = \text{median}(A + B, B, 2B)\}$

if $A > B$ *then*

$\{B = \text{median}(C, B, 2A)\}$

$C \leftarrow 2A$

else

$\{B = \text{median}(A + B, B, 2B)\}$

$C \leftarrow 2B;$

$\{B = \text{median}(A + B, B, C)\}$

$A \leftarrow A + B$

$B = \text{median}(A, B, C)$