Homework 3 Solution — Spring 2024

1

a (+6)

$S \to T \mid \varepsilon$
$T \to TV \mid V$
$V \to () \mid \{\} \mid [] \mid (T) \mid \{T\} \mid [T]$

other solutions are possible.

b (+6)

$S \Rightarrow T \Rightarrow TV \Rightarrow VV \Rightarrow \{T\}V \Rightarrow \{V\}V \Rightarrow \{[]\}V \Rightarrow \{[]\}()$

-1: skipped step.

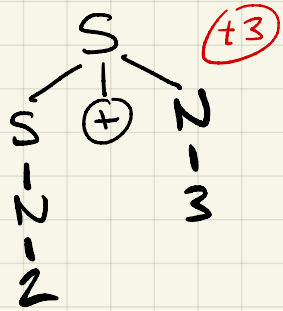
S: accounts for the empty string.
T: introduces side-by-side.
V: handles nested parts.

derivation doesn't need to be leftmost or rightmost.

2

a (+3)

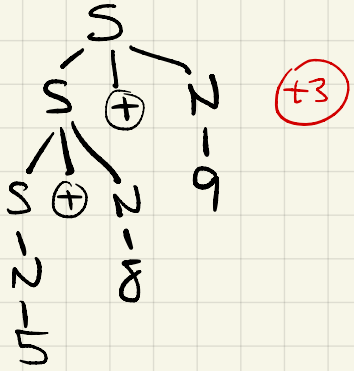
$S \Rightarrow S+N \Rightarrow N+N \Rightarrow 2+N \Rightarrow 2+3$



(+3)

b (+3)

$S \Rightarrow S+N \Rightarrow S+N+N \Rightarrow N+N+N \Rightarrow 5+N+N \Rightarrow 5+8+N \Rightarrow 5+8+9$

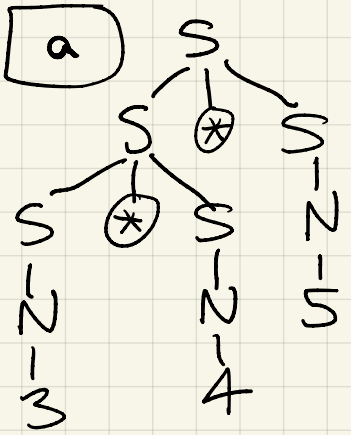


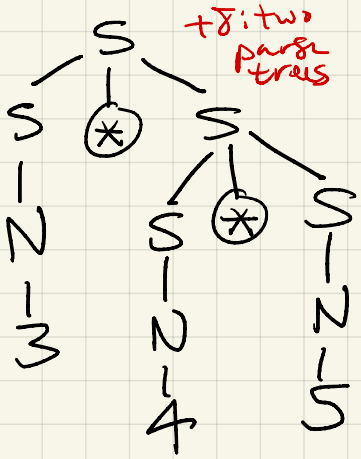
(+3)

-1.5: derivation isn't leftmost

3

a





+8: two parse trees

b yes, it's ambiguous. (+4)

We just drew two different parse trees of the same sentence.

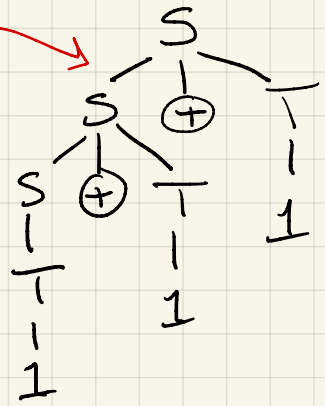
4

Here, ⊕ is left-associative. (+4)

Consider 1+1+1:

S
⇒ S+T
⇒ S+T+T
⇒ T+T+T
⇒ 1+T+T
⇒ 1+1+T
⇒ 1+1+1.

(+4)



Notice the tree is "left-weighted".

5

a S → l | o | m | f | g | ll | oo | mm | ff | gg | lSl | oSo | mSm | fSf | gSg (+8)

- l | o | m | f | g: middle of odd-length sentences
- ll | oo | mm | ff | gg: middle of even-length ones
- lSl | oSo | mSm | fSf | gSg: onion layers of matching terminals.

b (+8)

S
⇒ lSl
⇒ loSol
⇒ lolol //

* note: no ε possible.

6

a (+5)

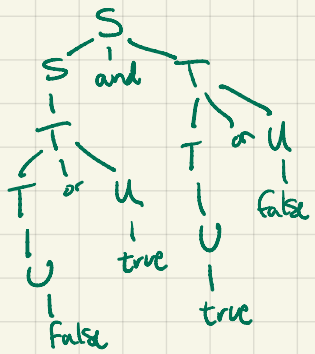
S → S and T | T
T → T or U | U
U → true | false

alternative: right associative.

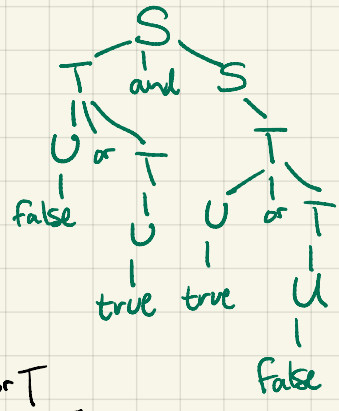
S → T and S | T
T → U or T | U
U → true | false

b (+5)

S
⇒ S and T
⇒ T and T
⇒ T or U and T
⇒ U or U and T
⇒ false or U and T
⇒ false or true and T
⇒ false or true and T or U
⇒ false or true and U or U
⇒ false or true and true or U
⇒ false or true and true or false

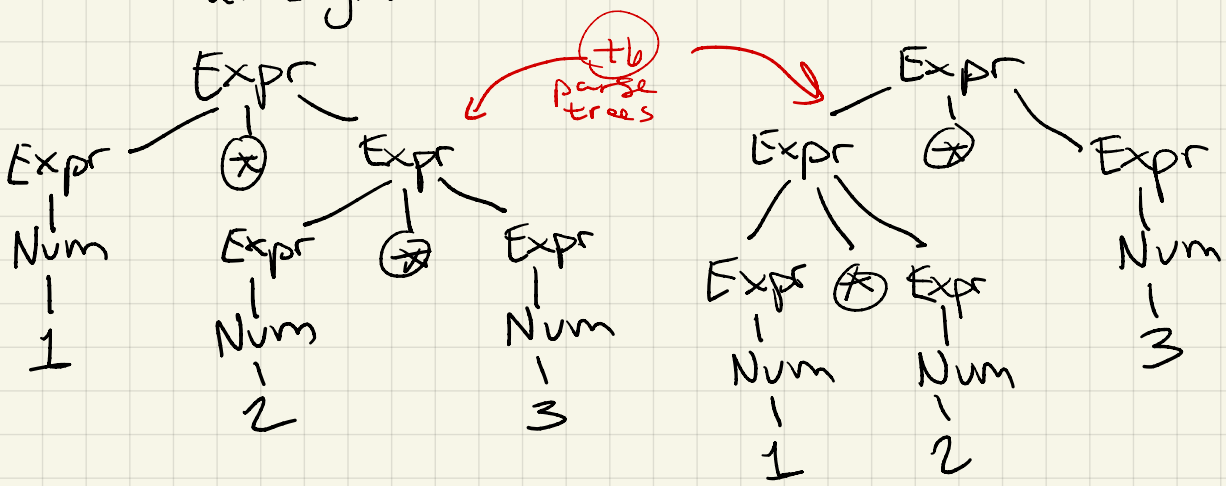


S
⇒ T and S
⇒ U or T and S
⇒ false or T and S
⇒ false or U and S
⇒ false or true and S
⇒ false or true and T
⇒ false or true and U or T
⇒ false or true and true or T
⇒ false or true and true or U
⇒ false or true and true or false



-2.5 : ambiguous. (doesn't have to have left assoc. operators)
-3 : introduced ( )'s

7 **a** 1*2*3 has two valid parse trees, so the language is ambiguous. (+2)



b Expr → Num * Expr | Num / Expr | Num

Num → (Expr) | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

+4: right-associative
+2: unambiguous, same language.
+2: parens handled correctly.

8

a yes. +3

b yes. +3

c no. +3

d yes. +3
