

# CS 314 Fall 2018

## Homework Assignment 1

### Answers

David Menendez

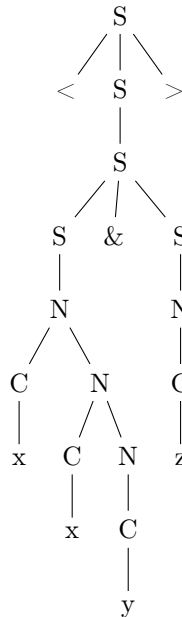
Due Tuesday, September 25, 3 AM

1. Consider the grammar  $G$  below. For each string, determine whether the string is part of  $L(G)$ . If it is in  $L(G)$ , prove it by giving a parse tree. If the parse is ambiguous, prove that by giving two parse trees. If the string is not in  $L(G)$ , explain why not (one sentence should be sufficient).

Start symbol  $S$   
 $S ::= '<' S '>' \mid S '&' S \mid N$   
 $N ::= C \mid C N$   
 $C ::= 'x' \mid 'y' \mid 'z'$

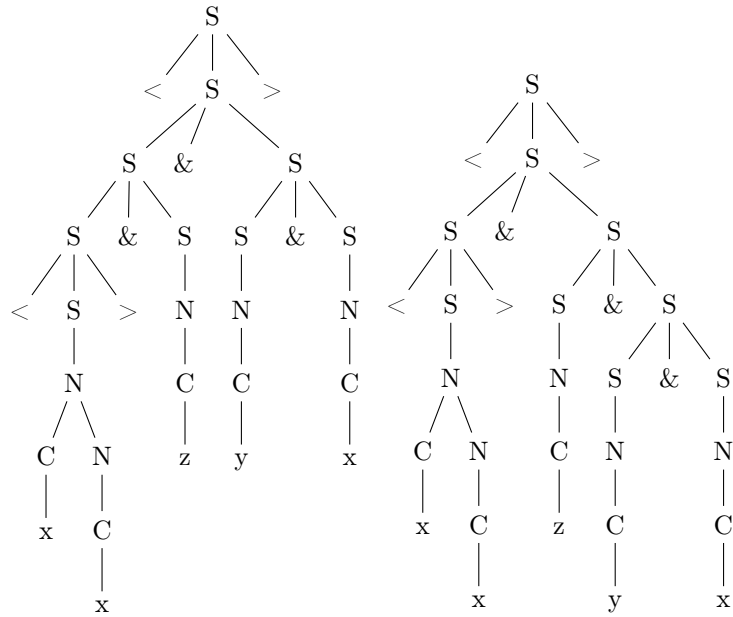
(a)  $<xy&z>$

Part of  $L(G)$ .



(b)  $\langle\langle xx\rangle\&z\&y\&x\rangle$

Part of  $L(G)$ . Ambiguous.

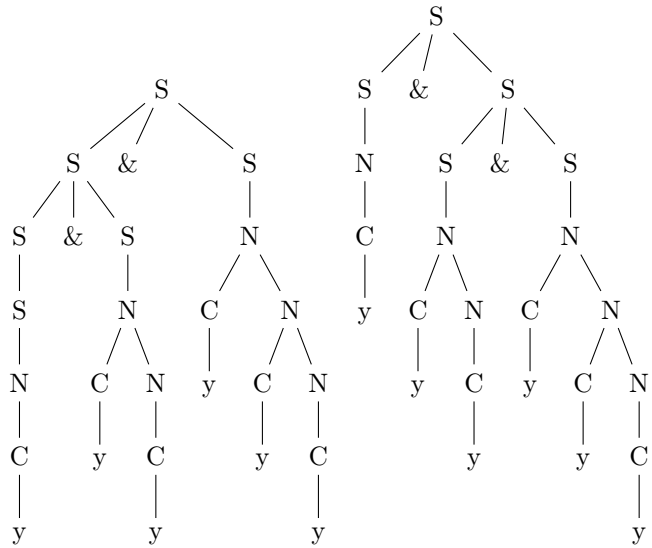


(c)  $\langle x\langle y\rangle\rangle$

Not part of  $L(G)$ .  $S$  can produce  $\langle y\rangle$ , but no rule has  $S$  immediately after  $S$ ,  $N$ , or  $C$ .

(d)  $y\&yy\&yyy$

Part of  $L(G)$ . Ambiguous.



(e)  $\langle\langle zxyz \rangle\rangle$

Not part of  $L(G)$ .  $S$  will produce  $\langle zxyz \rangle$ , but no rule produces  $\langle \langle S \rangle \rangle$ .

2. Rewrite  $G$  so that it is not ambiguous.

```
S ::= S & T | T
T ::= '<' S '>' | N
N ::= C | C N
C ::= 'x' | 'y' | 'z'
```

3. Write a regular expression for the language over alphabet  $\{a, b, c\}$  containing all strings with no more than two  $a$ 's.

$$(b|c)^*(a|\epsilon)(b|c)^*(a|\epsilon)(b|c)^*$$

4. Write a grammar for the same language. Indicate whether your grammar is ambiguous and briefly give your reasoning.

(Note: many possible answers. This is one.)

```
S    ::= BCs | BCs 'a' S2
S2   ::= BCs | BCs 'a' BCs
BCs  ::= e | BC BCs
BC   ::= 'b' | 'c'
```

This grammar is unambiguous. The  $BCs$  non-terminal associates strings of zero or more  $b$ 's and  $c$ 's to the left. Each production for the  $S$  and  $S2$  non-terminals is chosen by the number of  $a$ 's present in the string.