MA/CSSE 474 Homework #11 63 points

Solutions should be written clearly, or typed and printed (preferably double-sided).
2.1 means Exercise 1 from Chapter 2.

1. 11.11 (parse tree for URI)
2. (t-6-6) 11.12b (prove grammar correct)

> 11.12b **Note** that there are two things to show; **use induction for each one:**
> (a) Every string in *Bal* can be derived from the grammar ;
> (b) every string that can be derived from the grammar is in *Bal* (easiest to show something more general by induction on the length of the derivation and then use that to show this property).

3. (t-6-6) 11.13a (Show grammar ambiguous, then disambiguate)

4. (t-6) 11.14 (finiteness of set of strings whose derivations have length less than n)

Problem 4 previous questions and answers from Piazza:

Q: Any sort of help on this one? Are we just showing that there are a finite number of strings that have derivations of length n or less? Is this an induction proof...?

My answer: *Are we just showing that there are a finite number of strings that have derivations of length n or less?* Yes

Any sort of help on this one? One simple approach is to give an upper bound (does not have to be a tight upper bound) on the maximum length of a string that can be derived in n steps.

Is this an induction proof...? It could be, but you can be more informal if you wish.

5. 11.15 (if you are interested in compilers, you might want to try this one)

6. (t-6-3-6) 11.17 (Chomsky normal form)

Problem 6 previous questions and answers from Piazza:

Q: The *removeLong* algorithm that is part of *convertToChomsky* doesn't really make sense to me, especially step 4.

Is it saying that for each rule, you must add a rule for every character on the RHS of the string, and then link it to the next one (M2 to M3 and so on)? **A:** Yes, not only a new rule, but also a new nonterminal symbol.

Q: Also, if M2 -> M3 and M3 ->T2, then is it necessary or can you just do M2 -> TT2 since that's still in Chomsky form?
A: You can make simple shortcuts like this that work for special cases, as long as the result is in CNF, and the new grammar is equivalent to the old one.

7. 11.18 (convert specific grammars to CNF)

8. (t-6) 12.1b

9. (t-6) 12.1c

10. (t-6) 12.1d