Homework: Context-Free Grammar

Learning Objectives:

In this homework, we are going to exercise the following key knowledge points on the topic of context-free grammar (CFG)

1. understanding the relations of strings and grammars
2. performing derivations and constructing parse trees
3. determining and resolving ambiguity
4. designing a grammar to describe given string patterns

Instructions:

1. Total points: 40 pt
2. Early deadline: Feb 10 (Wed) 11:59 pm, Regular deadline Feb 12 (Fri) 11:59 pm (you can continue working on the homework till TA starts to grade the homework)
3. How to submit:
   - Submit your document to Canvas under Assignments, Homework 1
   - Please provide the complete solutions in one pdf file
   - You can write your solutions in latex or word and then convert it to pdf; or you can submit a scanned document with legible handwritten solutions

Questions:

1. (10 pt) [Grammar and strings] Given a string 010#0101 and the context free grammar G:
   $S \rightarrow AB$
   $A \rightarrow 0A0|1A1|\#B$
   $B \rightarrow 0B|1B|\epsilon$
   (a) (2 pt) What are the terminals and non-terminals of the grammar?
   (b) (2 pt) Give a leftmost derivation for the string.
   (c) (2 pt) Give a rightmost derivation for the string.
   (d) (2 pt) Give a parse tree for the string .
   (e) (2 pt) Write 2 strings that do not belong to the language $L(G)$ but use only the terminals from $L(G)$.
2. (10 pt) [Ambiguity] Consider the following grammar:
   - terminals: $x, y, z, >, <, 0, 1, (,)$, if, then, else

- non-terminals: S, F, B, T, E, N
- start symbol: S
- production rules:
  $S \rightarrow F|T\ N\ T$
  $F \rightarrow$if B then S|if B then S else S
  $B \rightarrow (T\ E\ T)$
  $T \rightarrow x|y|z|1|0$
  $E \rightarrow > \mid <$
  $N \rightarrow +|-|=$

(a) (4 pt) Draw two different parse trees for the string
if $(x > y)$ then if $(x < z)$ then $x = 1$ else $x = 0$.

(b) (2 pt) Modify the grammar to remove ambiguity.

(c) (2 pt) Draw the parse tree for the string using new grammar.

(d) (2 pt) Explain how your new grammar modifies the parse trees you drew in the first step to remove ambiguity.

3. (10 pt) [Grammar analysis and semantics] In the following, we redefine the grammar for the bitwise operations:

- terminals: 0, 1, not, and, or, xor
- non-terminals: E, T, F, V
- start symbol: E
- production rules:
  E→ E and T | E or T | T
  T → F xor T |not T | F
  F → 0|1

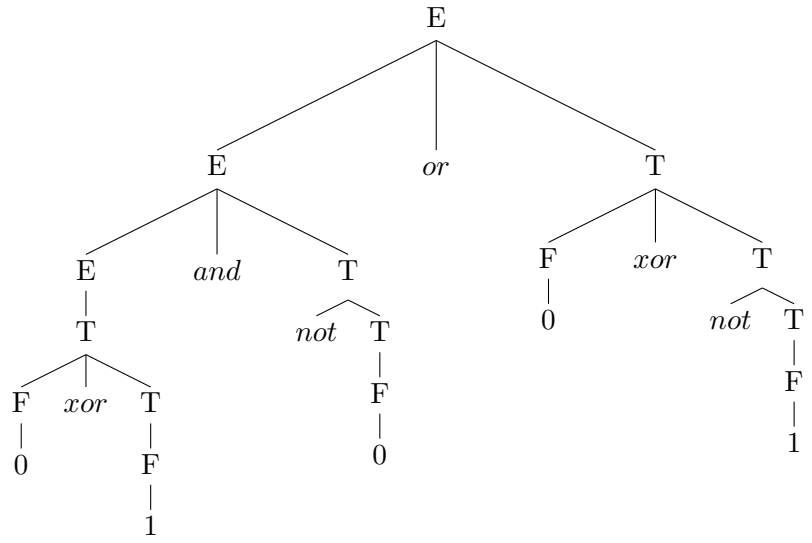
(a) (3 pt) What is the associativity of the operators "and", "or" and "xor"? Explain why.

(b) (3 pt) What is the precedence of "not","or", "and" and "xor"? Explain why.

(c) (4 pt) Consider the parse tree given below and answer the following questions.

i. (1 pt) What is the value of the string generated by the parse tree? (for the single operations "and", "or", "not" and "xor", use their usual semantics)

ii. (3 pt) Explain how the value of the string is generated step-wise.



4. (10 pt) [Designing Grammar] Design CFGs for the given languages:

   (a) (3 pt) Write a grammar that describes the strings $(0|1)^*(23)^+$.

   (b) (3 pt) Write a grammar that describes the paired brackets [and] . See the examples of valid strings: [], [[]], [][], [[][]] ...

   (c) (4 pt) Write a grammar that describes all the palindrome strings on the alphabet (terminals) of a, b and c. See the examples of valid strings, aca, bab, abba, acccca ...