Homework 3  10 points

Note, these exercises may be done in groups of one, two, or three. Working with someone else is strongly recommended. If more than one person is involved, list all the names on ONE set of answers.

Syntax

1. Given the regular expression (a|b|c*)bb use Thompson’s construction to produce a NFA (non-deterministic finite automata).

2. For the NFA(non-deterministic finite automata) to the right, use the subset construction (discussed in class) to create a DFA(deterministic finite automata)

3. Consider the grammar below where ⊕, ⊗, and© are some unspecified binary operators.

E →T⊕E | E⊗T | T
T →F⊗T |F
F →(E) |a

i. What are the terminals of this grammar? The non-terminals?
ii. What kind of associativity does ⊕ have?
iii. What kind of associativity does ⊗ have?
iv. What kind of associativity does © have?
v. What operation has higher precedence between ⊕ © ⊗ () or are they equal?

4. In English, what language is generated by the following grammar? Be as precise as possible. ε indicates the empty string.

T →aTb|bTa|ε

5. Demonstrate the grammar is ambiguous.

S → aS |SaS | ε

6. Consider the following grammar

Type → Id | Type : Type | Type * Type
Id → a|b|c|d

a. Demonstrate that the grammar is ambiguous.
b. Are strings: a:b , a:b:c:d , a**b in the language generated by Type?
c. Specify the same language using a regular expression
d. Specify the same language using a regular grammar.

7. In English, describe what language is generated by the following grammar? Be as precise as possible. S → SaS | b
8. In English, describe what language is generated by the following grammar? Be as precise as possible. \[ S \rightarrow SSS|a|ab \]

9. In English, describe what language is generated by the following grammar? Be as precise as possible. (\( \varepsilon \) indicates the empty string.) \[ S \rightarrow aSbS | bSaS | \varepsilon \]

10. Write a context free grammar to recognize \( L = \{a^{2n}b^m | 2n \leq m \leq 3n\} \) (Thus, aaabbbbb or aabbbbbbb are legal, but aab is not.)

11. Consider the grammar \( G \) represented by the four tuple = \( (\{E,T,F\}, \Sigma=\{+,\times, (,), a\}, P,E) \) where \( P \) is the set of productions
   
   \[
   \begin{align*}
   E & \rightarrow E + T | T \\
   T & \rightarrow T \times F | F \\
   F & \rightarrow (E) | \text{NUM}
   \end{align*}
   \]
   For this grammar, draw a parse tree for \( 3 *(4 + 5) + (6+7) \)

12. Using the CFG grammar for expressions (in the previous problem), add operators \( \% \) for mod and \( ^\) for exponentiation. Recall that mod is left-associative (like division) and that power is right associative. Test out your grammar to verify that it works.

13. Consider the syntax diagram below in which ovals represent terminals and rectangles represent non-terminals. Note, the items in rectangles will be referenced in your answer, but not defined. Give the equivalent grammar of the syntax diagram below using BNF

![Syntax Diagram](image-url)