Chapter 4 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.
   \[\begin{align*}
   E &\rightarrow T \oplus E \mid E \otimes T \mid T \\
   T &\rightarrow F \otimes T \mid F \\
   F &\rightarrow (E) \mid a
   \end{align*}\]
   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. \(\varepsilon\) indicates the empty string.
   \[T \rightarrow aTb|bTa|\varepsilon\]

5. Here is a simple grammar for conditionals in a language. (I’ve turned some non-terminals into terminals for simplicity.) Demonstrate the grammar is ambiguous.
   \[\begin{align*}
   \text{Statement} \rightarrow &\text{ Conditional} \\
   \text{Conditional} \rightarrow &\text{ IF TEST THEN Statement} \\
   \end{align*}\]

6. Consider the following grammar
   \[\begin{align*}
   \text{Type} &\rightarrow \text{Id} \mid \text{Type} : \text{Type} \mid \text{Type} \ast \text{Type} \\
   \text{Id} &\rightarrow a|b|c|d
   \end{align*}\]
   a. Demonstrate that the grammar is ambiguous.
   b. Are strings: \(a:b\), \(a:b:c:d\), \(a**b\) in the language of 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. \((\varepsilon\) indicates the empty string.) \[ B \rightarrow BB \mid (B) \mid \varepsilon \]

8. Write a context free grammar to recognize \(L = \{a^n b^m \mid 2n \leq m \leq 3n\}\) (Thus, aaabbbaa or aabbbbbb are legal, but aab is not.)

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

10. 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.

11. 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