HW1
50pts + 5pts extra credit
This is a team assignment. Form teams, and submit a pdf in Submitty.
Posted Tuesday, September 4, 2018
Due Tuesday, September 11, 2018

**Problem 1** (8pts). [Modified from Scott] Errors in a computer program can be classified according to *when* they are detected and, if they are detected at compile time, *what part* of the compiler detects them. Using C++ or Java, give an example of each of the following:

(a) (2pts) A lexical error, detected by the scanner
(b) (2pts) A syntax error, detected by the parser
(c) (2pts) A static semantic error, detected by semantic analysis
(d) (2pts) A dynamic semantic error, detected by code generated by the compiler (or detected by the virtual machine, in the case of Java)

**Problem 2** (12pts). [Modified from Aho, Sethi, Ullman] Describe in English the languages denoted by the following regular expressions.

a) (2pts) $a(a|b)^*$
b) (2pts) $((\varepsilon | a)b)^*$
c) (2pts) $(a|b)^*a(a|b)(a|b)$
d) (2pts) $a^*ba^*ba^*ba^*$
e) (4pts) $(aa|bb)^*((ab|ba)(aa|bb)^*(ab|ba)(aa|bb)^*)^*$

Note: Your description should be a high level characterization — one that would still make sense if we were using a different regular expression for the same language. For example, $(ba^*ba^*)^*$ should be described as “All strings of a’s and b’s, beginning with b and having even number of b’s.” not as, for example, “The string of b followed by any number of a’s followed by a b followed by any number of a’s, repeated any number of times.”.

**Problem 3** (15pts). [From Aho, Sethi, Ullman]
Consider the grammar

$$S \rightarrow 0 S 1 S \mid 1 S 0 S \mid \varepsilon$$

(a) (3pts) Show that this grammar is ambiguous by constructing two different leftmost derivations for the sentence 0101.
(b) (3pts) Construct the corresponding rightmost derivations.
(c) (3pts) Construct the corresponding parse trees.
(d) (6pts) The grammar generates strings with equal number of 0’s and 1’s. Does it generate *all* such strings? (You are not required to write a formal proof. If you answer YES, write a brief justification. If you answer NO, show a string that cannot be generated by the grammar.)
Problem 4 (15pts). [Modified from Aho, Sethi, Ullman] The following is an ambiguous grammar for expressions with one unary operator $*$ and $n$ binary infix operators, $\theta_1, \theta_2, \ldots, \theta_n$, at $n$ different levels of precedence:

$$E \rightarrow E \theta_1 E \mid E \theta_2 E \mid \ldots \mid E \theta_n E \mid E^* \mid (E) \mid \text{id}$$

(a) (5pts) Show that the grammar is ambiguous.

(b) (10pts) Construct an equivalent unambiguous grammar such that unary operator $*$ is left associative and binary operators $\theta_1, \theta_2, \ldots, \theta_n$ are all right associative. Operator $*$ has the highest precedence. $\theta_n$ takes precedence over $\theta_{n-1}$, $\theta_{n-1}$ takes precedence over $\theta_{n-2}$, etc.

Problem 5 (5pts extra credit. NO PARIAL CREDIT. Only full credit for a complete solution will be assigned.). [Modified from Sethi] The following grammar generates numbers in binary notation. $C$ is the start symbol.

(1) $C \rightarrow C 0 \mid A 1 \mid 0$

(2) $A \rightarrow B 0 \mid C 1 \mid 1$

(3) $B \rightarrow A 0 \mid B 1$

(a) (0pts) Warmup: Construct a derivation that generates the binary notation of 21.

(b) (1.5pts) Prove that the generated numbers are multiples of 3. (**Hint: Structural induction.**)

(c) (1.5pts) Prove that all such numbers (i.e., nonnegative numbers that are multiples of 3) are generated by the grammar. (**Hint: Strong induction on string length, contradiction.**)

(d) (2pts) Write a regular grammar $G$ such that the numbers generated by $G$ are multiples of 7 and all such numbers (i.e., nonnegative multiples of 7) are generated by $G$. 