Midterm

Mark all correct answers in each of the following questions.

4. For any $n \geq 0$, let L_n be the language of all legal expressions in two types of parentheses, [] and {}, of nesting level up to n. (For example, the four expressions [}, [{]}, [[], }{ are illegal. The expressions []{}[], [{}[]]{}, [[]]{[{}]} are of nesting levels 1, 2 and 3, respectively.) $L_{[]n}$ denotes the susbet of L_n consisting of expressions in [and] only. Define $L_{\{\}n}$ analogously. We agree that $L_0 = L_{[]0} = L_{\{\}0} = \{\varepsilon\}$. We would like to have regular definitions $l_n, l_{[]n}, l_{\{\}n}$ for the languages $L_n, L_{[]n}, L_{\{\}n}$, respectively.

   (a) After $l_1, l_2, \ldots, l_n$ have been defined, we may take
   $l_{n+1} \to l_1 l_n \mid l_2 l_{n-1} \mid \ldots \mid l_n l_1$.

   (b) After $l_1, l_2, \ldots, l_n$ have been defined, we may take
   $l_{n+1} \to l_n \mid [l_n^*] \mid \{l_n^*\}$.

   (c) After $l_1, l_2, \ldots, l_n$ have been defined, we may take
   $l_{n+1} \to (l_n \mid [l_n] \mid \{l_n\})^*$.

   (d) After $l_1, l_2, \ldots, l_n$ have been defined, we may take
   $l_{n+1} \to (l_n^* \mid ([l_n])^* \mid (\{l_n\})^*)^*$.

   (e) After $l_{[]1}, l_{[]2}, l_{[]3}, l_{\{\}1}, l_{\{\}2}, l_{\{\}3}$ have been defined, we may take
   $l_{n+1} \to l_{[]3} \mid l_{\{\}3} \mid l_{[]2} l_{\{\}1} \mid l_{\{\}2} l_{[]1} \mid l_{[]1} l_{\{\}2} \mid l_{\{\}1} l_{[]2} \mid l_{[]1} l_{\{\}1} l_{[]1} \mid l_{\{\}1} l_{[]1} l_{\{\}1}$.

5. (a) If a context-free grammar G contains the rule $A \to AbA$, then it is necessarily ambiguous.

(b) The following grammar (in which $\circ$ represents a certain binary operation) is used to describe a certain class of expressions:

$$E \rightarrow E\underline{\circ}E \mid \underline{a} \mid \underline{b} \mid \underline{c}$$

If the operation $\circ$ is associative (for example, addition or multiplication of numbers), then the grammar is unambiguous. However, if $\circ$ is non-associative (for example, subtraction or division), then the grammar is ambiguous (because, for example, $a-b-c$ may be derived from two different parse trees, one of which would imply that we indeed refer to the expression $a-b-c$, while the other would imply that we actually mean $a-(b-c)$).

(c) The grammar given by the rules
$stmt \;\rightarrow\;$ tripleIf exp $stmt$ else $stmt$ elselse $stmt$ | $unconditionalStmt$
is unambiguous. (You should interpret the right-hand side of the first rule as follows: exp is an integer expression. If it assumes a positive value the first statement is executed, if negative – the second statement, if 0 – the third statement. Ignore possible problems due to rules relating to exp, $stmt$ and $unconditionalStmt$. Here we care only about ambiguity that is due to the given pair of rules.)

(d) Let G be a context-free grammar. If A is known to be a useful letter in G and
$A \rightarrow AbBBaA \mid BaB$
are the only rules for A in G, then B is also a useful letter.

6. Let $d \geq 2$ be an arbitrary fixed integer. A non-negative integer n is *super-even* in base d (for the purposes of this question) if all permutations of its base d expansion form even numbers in base d. (For example, in base 10 the number 626 is super-even.) The number is *super-odd* if all these permutations form odd numbers. A number is *evenish* if there exists a permutation forming an even number, and *oddish* if there exists a permutation forming an odd number. You may check, for example, that the regular definition

$$\begin{array}{lcl} evenDigit & \rightarrow & 0 \mid 2 \mid \ldots \mid d_{\text{even}} \\ superEven & \rightarrow & evenDigit^{+} \end{array}$$

where d_{even} denotes the largest even digit in base d, defines all expansions of super-even numbers in case d itself is even.

(a) The regular definition

$$\begin{array}{lcl} \mathit{oddDigit} & \to & 1 \mid 3 \mid \ldots \mid d_{\text{odd}} \\ \mathit{superOdd} & \to & \mathit{oddDigit}\ (\mathit{oddDigit}\ \mathit{oddDigit})^* \end{array}$$

where d_{odd} denotes the largest odd digit in base d, defines all expansions of super-odd numbers in case d itself is odd.

(b) If d is odd, then we cannot decide within one reading of the digits of a number whether there is a permutation of those digits that represents an even number. In particular, we cannot provide a regular definition for the set of all expansions of evenish numbers.

(c) If d is even, then we can construct a DFA with two states that will recognize base d expansions of evenish numbers. (The automaton is supposed to deal only with legal inputs – non-empty sequences of base d digits.)

(d) If d is odd, then we can construct a DFA with two states that will recognize base d expansions of oddish numbers. (The automaton is supposed to deal only with legal inputs – non-empty sequences of base d digits.)

Solutions

4. (a) By concatenating two expressions in parentheses, we obtain an expression whose nesting level is the maximum between the levels of nesting of the two expressions. Hence the suggested regular expression describes only words belonging to L_n.

   (b) The suggested regular expression does not describe concatenations of several expressions of nesting level $n+1$, for example $[^{n+1}]^{n+1}[^{n+1}]^{n+1}$.

   (c) The suggested regular expression describes only concatenations of expressions of the forms $\alpha, [\alpha]$ and $\{\alpha\}$, where α is an expression of nesting level n. Hence the language described by the regular expression is contained in L_{n+1}. On the other hand, let $w \in$

L_{n+1}. If $w = w_1 w_2$ for some non-empty words w_1, w_2, where the number of occurrences of $[, \{$ in w_1 is equal to the number of occurrences of $], \}$, respectively, then $w_1, w_2 \in L_{n+1}$, and we show that w belongs to the language described by the suggested regular expression using induction on the length of w. Otherwise, either $w = \varepsilon$, or $w = [w_1]$ for some $w \in L_n$, or $w = \{w_1\}$ for some $w \in L_n$R; all three cases are clearly covered by the definition.

(d) The suggested regular expression clearly describes the same language as that of the preceding part (albeit in a more cumbersome form).

(e) The language described by the suggested regular expression does not contain, for example, the word $[^3]^3\{^3\}^3$.

Thus, (c) and (d) are true.

5. (a) We may clearly produce the string $AbAbA$ from A in two distinct ways. If the non-terminal A is useful, then by taking a string $\alpha A \beta$ and three words w_1, w_2, w_3 consisting of terminals only, with $S \overset{*}{\Longrightarrow} \alpha A \beta$, $\alpha \overset{*}{\Longrightarrow} w_1$, $A \overset{*}{\Longrightarrow} w_2$, $\beta \overset{*}{\Longrightarrow} w_3$, we see that $S \overset{*}{\Longrightarrow} w_1 w_2 b w_2 b w_2 w_3$ in two distinct ways. However, if A is useless then we cannot conclude that G is ambiguous. For example, the grammar defined by the rules

$$S \to SA \mid \varepsilon\,,$$
$$A \to AbA\,,$$

is clearly unambiguous.

(b) The grammar is ambiguous since the word $a \circ b \circ c$, for example, may be produced from S in two distinct ways. The "meaning" of the operation $\circ$ is irrelevant.

(c) Since we care only about possible ambiguity due to the presented rule, we may as well consider the grammar defined by the rules

$$S \to tSe_1Se_2S \mid \varepsilon\,.$$

(Here, S, t, e_1, e_2 represent *stmt*, tripleIf, else, elselse, respectively.) One shows inductively that each word in $L(G)$ has the same number of occurrences of each of the three terminals t, e_1, e_2. Also, each prefix of the word contains at least as many occurrences of t as occurrences of e_1 and at least as many occurrences of e_1 as occurrences of e_2. Thus, the e_1 and e_2 corresponding to each t are uniquely determined by the condition that this e_1 is the first such that the number of each of the three terminals t, e_1, e_2 encountered since t are the same. Similarly, the e_2 corresponding to e_1 is the first such that the number of each of the three terminals t, e_1, e_2 encountered since e_1 are the same. In particular, the parse tree corresponding to each word in $L(G)$ is unique.

(d) Since A is useful, we have $S \stackrel{*}{\Longrightarrow} \alpha A \beta \stackrel{*}{\Longrightarrow} w$ for some α, β, w, the latter of which consisting of terminals only. This implies that either $S \stackrel{*}{\Longrightarrow} \alpha AbBBaA\beta \stackrel{*}{\Longrightarrow} w$ or $S \stackrel{*}{\Longrightarrow} \alpha BaB\beta \stackrel{*}{\Longrightarrow} w$. In either case we see that B is useful.

Thus, (c) and (d) are true.

6. Let us first characterize the numbers of the various types, defined in the question, for even and odd bases d. Note that, in any base, a number is evenish (oddish, resp.) iff it is not super-odd (super-even, resp.). If d is even, then an expansion represents a super-even number if all digits are even and an evenish number if at least one of its digits is even. If d is odd, then an expansion represents a super-even number if the number of odd digits is even, which is the case iff the number is even.

(a) The suggested expression describes all sequences of odd length of odd digits. Thus it fails to describe those expansions that contain also (any number of) even digits.

(b) To decide whether an expansion represents an even number, we just need to know whether the number of odd digits in this expansion is even. This is readily accomplished with a DFA.

(c) The DFA

$$M = (\{\mathsf{EVENISH}, \mathsf{ODDISH}\}, \{0, 1, \ldots, d-1\}, \delta, \mathsf{ODDISH}, \{\mathsf{EVENISH}\}) ,$$

where the transition function is given by

$$\delta(q, \sigma) = \begin{cases} \mathsf{ODDISH}, & q = \mathsf{ODDISH},\ \sigma = 1, 3, \ldots, d-1, \\ \mathsf{EVENISH}, & \text{otherwise}, \end{cases}$$

does the required task.

(d) The DFA

$$M = (\{\mathsf{EVENISH}, \mathsf{ODDISH}\}, \{0, 1, \ldots, d-1\}, \delta, \mathsf{EVENISH}, \{\mathsf{ODDISH}\}) \ ,$$

where

$$\delta(q,\sigma) = \begin{cases} \mathsf{EVENISH}, & q = \mathsf{EVENISH}, \quad \sigma = 0, 2, \ldots, d-2, \\ \mathsf{EVENISH}, & q = \mathsf{ODDISH}, \quad \sigma = 1, 3, \ldots, d-1, \\ \mathsf{ODDISH}, & q = \mathsf{EVENISH}, \quad \sigma = 1, 3, \ldots, d-1, \\ \mathsf{ODDISH}, & q = \mathsf{ODDISH}, \quad \sigma = 0, 2, \ldots, d-2, \end{cases}$$

does the required task.

Thus, (c) and (d) are true.