SOLUTIONS

There are 125 points total. (So first exam is 20% and this is 25%.)

1. (45 pts.) Indicate whether true or false. Beware of guessing:

correct answer +5pts. incorrect answer −3pts. no answer 0pts

(a) T Every finite set of strings is a CFL.

(b) F The language $\{a^n b^n c^n | n > 0\}$ can be recognized by a (1-stack) PDA.

(c) T A PDA with two stacks can recognize more languages than a standard 1-stack PDA.

(d) T If L is Turing-decidable, then $\overline{L}$ is also Turing-decidable.

(e) F A Turing machine with two tapes can recognize more languages than a standard 1-tape Turing machine.

(f) T The language $\{a^n b^n c^n d^n | n > 0\}$ is Turing-recognizable.

(g) T L is Turing-decidable when L is the set of strings of digits that represent primes; that is, $L = \{2, 3, 5, 7, 11, 13, \ldots\}$. ($n$ is a prime if its only positive integer divisors are itself and 1.)

(h) T There exists a Turing machine which can decide if two DFAs are equivalent; that is, whether or not they recognize the same language.

(i) F There exists a Turing machine M which can decide if a Turing machine will loop on a given input; that is, M's input is a description of a machine, say T, and a string, say w, and M accepts the input if T does loop on w and M rejects the input if T does not loop on w.

SOLUTIONS

2. (25 pts.) Prove that, if L and M are CFLs, then so is $L \cup M$.

Ans. Suppose we have context free grammars for L and M with start symbols S_L and S_M and with no variable symbols in common. Define a new grammar whose start symbol is S and whose rules are $S \to S_L \mid S_M$ and union of the rules for L and M. Clearly, when we apply either $S \to S_L$ or $S \to S_M$, from then on we are working either in the grammar for L or in the grammar for M, respectively.

If you'd like a more formal proof: Let $(V_L, \Sigma_L, R_L, S_L)$ be a CFG that generates L and let $(V_M, \Sigma_M, R_M, S_M)$ be a CFG that generates M, where V_L and V_M are chosen so that they are disjoint and so that $V' = V_L \cup V_M$ is disjoint from $\Sigma = \Sigma_L \cup \Sigma_M$. Let S be a symbol that is not in V' or Σ Let $V = V' \cup \{S\}$ and $R = R_L \cup R_M \cup \{S \to S_L \mid S_M\}$. Consider the CFG (V, Σ, R, S). In a derivation, the first substitution will replace the string S with either S_L or S_M. In the first case, we can obtain any string in L. In the second case, we can obtain any string in M. Hence the CFG generates $L \cup M$.

3. (30 pts.) Let $L = \{a^n b c^n \mid n \geq 0\}$.

(a) Construct a context free grammar to generate the language.

Ans. The start symbol is S. There are only two rules: $S \to b$ and $S \to aSc$, which can also be written $S \to b \mid aSc$.

(b) Construct a PDA to recognize the language.

Ans. It's hard to draw with the software I have, so I'll describe it. The states are $q_1, \ldots, q_4$. The start state is q_1 and the accept state is q_4. There is one edge out of q_1, labeled $\epsilon : \epsilon \to \$$ and going to q_2. There is a loop from q_2 to itself labeled $a : \epsilon \to a$. There is an edge from q_2 to q_3 labeled $b : \epsilon \to \epsilon$. There is a loop from q_3 to itself labeled $b : a \to \epsilon$. There is an edge from q_3 to q_4 labeled $\epsilon : \$ \to \epsilon$.

4. (25 pts.) Suppose that both L and $\overline{L}$ are Turing-recognizable. Either (a) prove that L must be Turing-decidable, or (b) give an example of such an L which is not Turing-decidable.

Ans. (b) is impossible and (a) is true. It is the second part of the proof of Theorem 4.16 on pages 167–168. See the book for details. (You need to have the details on your exam.)

END