3515ICT Theory of Computation

Take-home test 2 answers

Each question is worth 4 marks.

1. Convert the following NFA to a DFA, using the subset construction algorithm...

	a	b
$\rightarrow$ * 1245	3456	5
* 3456	456	25
5	6	5
* 456	456	5
* 25	36	5
* 6	$\emptyset$	$\emptyset$
* 36	$\emptyset$	2
$\emptyset$	$\emptyset$	$\emptyset$

2. Construct the minimal DFA for the following DFA using the state equivalence algorithm...

	a	b
$\rightarrow$ 1	2	35
2	4	35
35	46	7
*46	46	35
*7	6	7

3. Use the pumping lemma for regular languages to prove that following language is not regular:

$$L_1 = \{ a^n b^{2n} \mid n \geq 0 \}$$

Proof. Suppose L_1 is regular. Let p be the pumping lemma constant for L_1. Let $a^p b^{2p} \in L_1$. By the pumping lemma, $s = xyz$ with $|xy| \leq p$, $|y| > 0$ and, for all $k \geq 0$, $xy^k z \in L_1$. Then $y = a^n$ with $0 < n \leq p$. Therefore, $xz = xy^0 z = a^{p-n} b^{2p}$, and $p - n < p$. By the pumping lemma, $xz \in L_1$. But xz does not contain twice as many b's as a's, so $xz \notin L_1$. This is a contradiction. So L_1 is not regular.

4. Give context-free grammars that generate the following languages.

   (a) $\{ a^m b^n c^n \mid m, n \geq 0 \}$
   $S \rightarrow AX$
   $A \rightarrow \varepsilon \mid aA$
   $X \rightarrow \varepsilon \mid bXc$

   (b) $\{ a^i b^j c^k \mid i = j + k \}$
   $S \rightarrow X \mid aSc$
   $X \rightarrow \varepsilon \mid aXb$

5. Consider the following grammar G over the alphabet $\{a, b, c\}$.

   $S \rightarrow c \mid aS \mid aSbS$

   (a) Informally describe $L(G)$.

   Answer 1. $L(G)$ is the set of strings in $\{a, b, c\}^*$ such that every prefix contains at least as many a's as b's and every (possibly empty) maximal substring of a's is followed by a c.

   Answer 2. Interpreting a as "if-then", b as "else" and c as "atomic statement", $L(G)$ is the language of "if-then-else" statements.

   (b) Prove that G is ambiguous.

   The string $aacbc$ has the following two distinct leftmost derivations:
   $S \rightarrow aS \Rightarrow aaSbS \Rightarrow aacbS \Rightarrow aacbc$
   $S \rightarrow aSbS \Rightarrow aaSbS \Rightarrow aacbS \Rightarrow aacbc$

   The same string also has two distinct parse trees, corresponding to the two leftmost derivations.

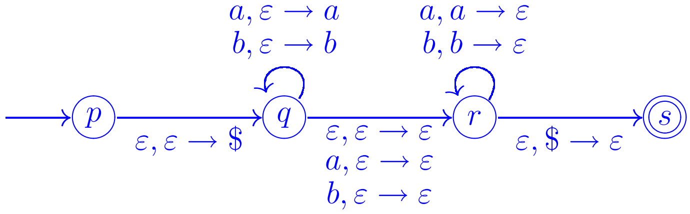
   (c) Give an unambiguous grammar for $L(G)$.

   $S \rightarrow c \mid aS \mid aTbS$
   $T \rightarrow c \mid aTbT$

6. Give pushdown automata that recognise the following languages.

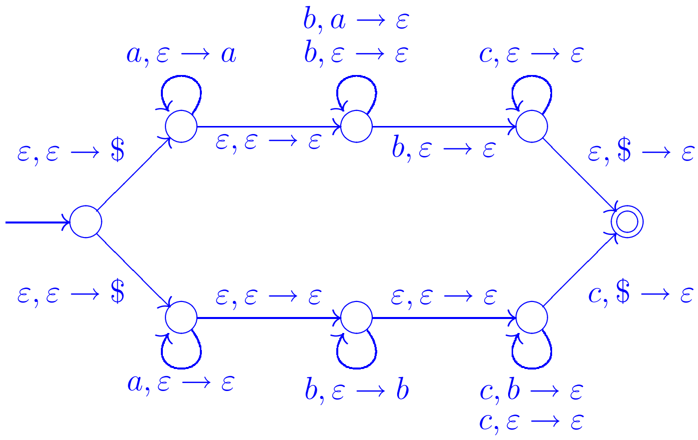
   (a) $\{\, w \in \{a, b\}^* \mid w = w^R \,\}$

   Because $|w|$ may be odd, this language is different from the one recognised by the PDA in Sipser, Example 2.18.



   (b) $\{\, a^i b^j c^k \mid i < j \text{ or } j < k \,\}$

   The top path recognises the first case and the bottom path the second case.

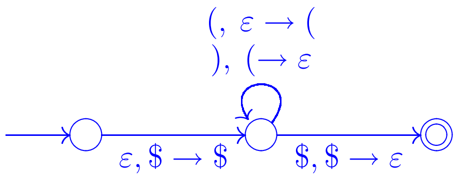


7. (a) Give an example of a language that is not deterministic, *i.e.*, that cannot be recognised by any deterministic PDA.

   There are many such languages, *e.g.*, $\{ ww^R \mid w \in \{a,b\}^* \}$ or $\{ w \in \{a,b\}^* \mid w = w^R \}$.

   (b) Give a deterministic PDA that recognises the language of balanced parenthesis strings.

   With all such questions, we may assume the input is followed by some "end of input" symbol, normally \$. With this question, push left parentheses and match right parentheses against left parentheses.



   (This question was too easy!)

8. Use the pumping lemma for context-free languages to prove the following language is not context-free:

$$L_2 = \{ a^{n^2} \mid n \geq 1 \}$$

Proof. Suppose L_2 is context-free. Let p be the pumping lemma constant for L_2. Choose $a^{p^2} \in L_2$. By the pumping lemma, $s = uvwxy$ with $|vwx| \leq p$, $|vy| > 0$ and, in particular, $uv^2xy^2z \in L_2$. Suppose $|vy| = n \leq p$. Then $|uv^2xy^2z| = p^2 + 2n \leq p^2 + 2p < (p+1)^2$. But there is no square between p^2 and $(p+1)^2$. So $uv^2xy^2z \notin L_2$. This is a contradiction, so L_2 is not context-free.

9. (a) Suppose that L is a context-free language and that R is a regular language. Prove that $L \setminus R$ is a context-free language.

   This is easy. $L \setminus R = L \cup \overline{R}$. But the complement of a regular language is regular, and the intersection of a context-free language and a regular language is context-free (Sipser, Problem 2.18a), so $L \setminus R$ is context-free.

   (b) Give a PDA that recognises the set of palindromes in $\{a,b\}^*$ such that the number of a's is a multiple of 3.

   Solution 1. First construct a grammar that generates this language. Here, T generates even-length strings in the language and U generates odd-length strings in the language.

   $S \longrightarrow T \mid U$
   $T \longrightarrow \varepsilon \mid bTb \mid aT_1a$
   $T_1 \longrightarrow bT_1b \mid aT_2a$
   $T_2 \longrightarrow bT_2b \mid aTa$
   $U \longrightarrow bUb \mid aU_2a$
   $U_2 \longrightarrow bU_2b \mid a \mid aU_4a$
   $U_4 \longrightarrow bU_4b \mid aU_6a$
   $U_6 \longrightarrow bU_6b \mid aU_2a$

   (This grammar could have errors and there could be a simpler grammar.)

Then apply the very simple grammar to PDA construction from Sipser, Lemma 2.21, omitted here.

Solution 2. First construct a PDA that recognises the set of palindromes in $\{a, b\}^*$. Next construct a DFA that recognises the strings in $\{a, b\}^*$ such that the number of a's is a multiple of 3. Then combine the two automata using the solution to Sipser, Problem 2.18a. The details are omitted.

Solution 3. Construct the PDA directly. I suspect this solution is more difficult than either of the other two solutions.

10. Use the general transformation from a pushdown automaton to a context-free grammar to construct a context-free grammar that generates the language this automaton recognises.

    This is an exercise in applying the construction of Sipser, Lemma 2.27.

Rodney Topor