Models of Computation (Sheet #3)

Marius Gavrilescu

1. (a) Suppose the language is regular, and let p be the pumping length. We will take the string $s = 0^p10^p$. By the pumping lemma there are x, y, z such that $s = xyz$ where $|xy| \leq p$ and $|y| > 0$. Therefore $x = 0^a, y = 0^b, z = 0^{p-a-b}10^p$. The lemma also asserts $xy^iz \in L \forall i \geq 0$. But $xz = 0^{p-b}10^p$ with $b \neq 0$, which is not in L. This contradicts the lemma's assertion, so the language is not regular.

   (b) Similarly to the above part we'll let p be the pumping length and take $s = 0^{p!}1^{2 \cdot p!}$ as suggested by the hint. There will be some x, y, z as before, so $x = 0^a, y = 0^b, z = 0^{p!-a-b}1^{2 \cdot p!}$ with $a + b \leq p$. This means $b \leq p$, so $b|p!$. Therefore we can select $i = \frac{p!}{b} + 1$, in which case xy^iz is $0^{2 \cdot p!}1^{2 \cdot p!} \notin L$. This contradicts the lemma's assertion, so the language is not regular.

2. (a) Regular languages are closed under complement, so if L was regular then the set of palindromes would have to be regular as well. This is a contradiction, so L is not regular.

   (b) Take strings $s_i = 0^i1$. For any $i \neq j$, $s_i \not\equiv_L s_j$ because $s_i0^i \notin L$ while $s_j0^i \in L$. So there are infinitely many equivalence classes, which means L is not regular.

   (c) Similarly to 1(i) we will let p be the pumping length and take $s = 0^{p!}10^{2 \cdot p!}$.

3. (a) Take any word in the language. If it is in L_1, then it looks like $01^i01^j01^j$. We can take $x = 0, y = 1, z = 1^{i-1}01^j01^j$. xy^iz will be either $01^k01^j01^j$ with $k > 0$ or 001^j01^j, which are in $L_1 \cup L_2$.

   Otherwise, it is in L_2, we look at the first three digits. If any of the digits are 1, we select it as y because repeating it (or removing it) does not remove our string from L_2. If all of the digits are 0, we select any of them. Repeating it keeps our string in L_2, and removing it still keeps it in L_2 because the other two 0s are now consecutive.

   (b) Take strings $s_i = 0101^i01$ for $i > 1$. Now for any $i \neq j$ we have $s_i \not\equiv_L s_j$ because $s_i1^{i-1} \in L_1 \cup L_2$ while $s_j1^{i-1} \notin L_1 \cup L_2$.

   (c) No, because the Pumping Lemma is not a characterisation of regular languages. It holds for all regular languages, but also for some non-regular languages.

4. Empty set is accepted by $(\{A\}, \{0, 1\}, (A \to \epsilon), A)$.

   Palindromes are accepted by $(\{A\}, \{0, 1\}, (A \to \epsilon|0|1|0A0|1A1), A)$.

   Strings with at least three ones are accepted by $(\{A, B, C, D\}, \{0, 1\}, \mathcal{R}, D)$ where the rules are:
   $A \to \epsilon|0A|1A$
   $B \to A1A$
   $C \to B1A$
   $D \to C1A$

   Odd-length strings whose middle symbol is 0 are accepted by
   $(\{A\}, \{0, 1\}, (A \to 0|1A0|0A1|0A0|1A1), A)$

5. (a) $(\{A, B, C\}, \{a, b\}, \mathcal{R}, A)$ with rules:
   $A \to B|aAb$
   $B \to bbbC$
   $C \to \epsilon|bC$

   (b) $(\{A\}, \{a, b\}, \mathcal{R}, A)$ with rules:
   $A \to aAbb|Bab|Ba|C$
   $B \to \epsilon|aB$
   $C \to \epsilon|bC$

6. (a) Well-bracketing means that a string that is either empty or formed of an open paranthesis of one kind, followed by a well-bracketed string, followed of a closed paranthesis of the same kind as the first one, and followed by another well-bracketed string.

   Writing this as a CFG, $(\{A\}, \{(,), [,]\}, (A \to \epsilon|[A]|(A)|AA), A)$

   (b) $(\{A, B\}, \Sigma, \mathcal{R}, A)$ with rules:
   $A \to \epsilon|(B)|AA$
   $B \to \epsilon|[A]|BB$