Midterm Examination
CS 4232: Theory of Computation

Thursday 17 September 2015, Duration 40 Minutes

Matriculation Number: ________________

Rules: This test carries 20 marks and consists of 4 questions. Each questions carries 5 marks; full marks for a correct solution; a partial solution can give a partial credit. Use the backside of the page if the space for a question is insufficient.

Question 1 [**5 marks**]

Construct a complete deterministic finite automaton with as few states as possible which recognises the set $((\{00\} \cdot \{0\}^*) \cup (\{11\} \cdot \{1\}^*) \cup (\{22\} \cdot \{2\}^*))^*$. The alphabet is $\{0, 1, 2\}$. Recall that a finite automaton is deterministic and complete iff for every state q and every symbol a there is exactly one successor state $\delta(q, a)$ to which it can go.

Solution. The dfa is given as follows: The set of states is $\{s, q_0, q_1, q_2, r_0, r_1, r_2, p\}$ and the alphabet is $\{0, 1, 2\}$. The state-transition function δ uses in the following definition a symbol a (if needed) as the index of the state and b as the input symbol currently processed: $\delta(s, b) = q_b$; if $a = b$ then $\delta(q_a, b) = r_a$ else $\delta(q_a, b) = p$; if $a = b$ then $\delta(r_a, b) = r_a$ else $\delta(r_a, b) = q_b$; $\delta(p, b) = p$. Furthermore s is the starting state and $\{s, r_a, r_b, r_c\}$ is the set of accepting states. The states q_a differ from all r_b as the first are rejecting and the latter are accepting. The start state s is the unique state from which one can go within two but not within one step into an accepting state. The state p is the unique state from which one cannot go into an accepting state. Note that if $a \neq b$ then the states q_a and q_b differ as from q_a one goes on a into an accepting state but not from q_b; similarly for r_a and r_b. Furthermore, all states are reachable. Thus the dfa is minimal.

Recall the traditional form of the Pumping Lemma: Let $L \subseteq \Sigma^*$ be an infinite regular language. Then there is a constant k such that for every $u \in L$ of length at least k there is a representation $x \cdot y \cdot z = u$ such that $|xy| \leq k$, $y \neq \varepsilon$ and $xy^*z \subseteq L$.

Recall that a word w is a palindrome iff the mirror image of w^{mi} is equal to w; so 001100001100 and 01210 are palindromes while 0001 is not. Let $H = \{w \in 0^+1^+2^+1^+0^+ : w \text{ is a palindrome}\}$. Which of the following three choices is correct?

(a) H is regular and satisfies the Pumping Lemma;

(b) H is not regular but still satisfies the Pumping Lemma;

(c) H does not satisfy the Pumping Lemma and is thus not regular.

Prove your answer.

Solution. The correct choice is **(c)**.

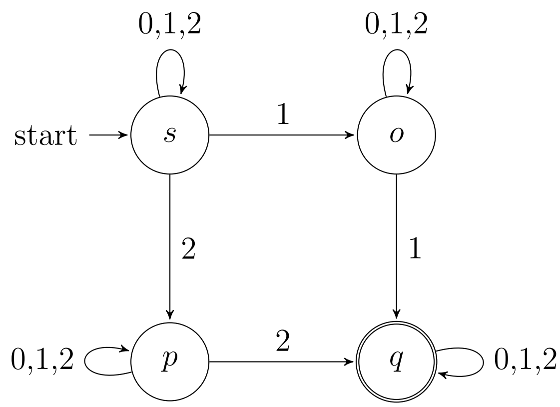
Assume that H satisfies the Pumping Lemma with constant k and consider the word $0^k1^k2^k1^k0^k$ which is in H. If H would be regular then there are x, y, z with $xyz = 0^k1^k2^k1^k0^k$, $y \neq \varepsilon$, $|xy| \leq k$ and $xz \in H$. Due to the length constraints, $x \in 0^*$ and $y \in 0^+$. Now $xz = 0^h1^k2^k1^k0^k$ for a number $h < k$ and is not a palindrome, thus $xz \notin H$ and the Pumping Lemma cannot be satisfied for H. Thus H cannot be a regular set as well.

Consider the context-free grammar

$$(\{S, T\}, \{0, 1, 2\}, \{S \to T1T1T | T2T2T, T \to TT|0|1|2|\varepsilon\}, S).$$

Is the language L generated by this grammar regular? If so, provide an non-deterministic finite automaton recognising L; if not, give a proof that the language is not regular.

Solution. The language is regular. A non-deterministic automaton for this language is given as follows:



The following $\{0, 1, 2, 3\}$-valued function F is defined by structural induction for all regular expressions:

- $F(\emptyset) = 0$, $F(\{\varepsilon\}) = 1$;
- $F(\{w_1, w_2, \ldots, w_n\}) = 2$ in the case that at least one of the w_m is a nonempty word – otherwise the previous case applies;
- $F((\sigma \cup \tau)) = \max\{F(\sigma), F(\tau)\}$;
- If $F(\sigma) = 0$ or $F(\tau) = 0$ then $F((\sigma \cdot \tau)) = 0$ else $F((\sigma \cdot \tau)) = \max\{F(\sigma), F(\tau)\}$;
- If $F(\sigma) \leq 1$ then $F(\sigma^*) = 1$ else $F(\sigma^*) = 3$.

In these definitions, it is always assumed that brackets are used to make the breaking down of expressions unique and that σ, τ are valid regular expressions using as constants $\emptyset$ and lists of finite sets of strings and as connectives $\cup$, $\cdot$ and *. Answer the following questions:

- What is $F(((\{00, 11\}^* \cdot \emptyset) \cup \{00, 11, 22\}))$?
- For which regular expressions does it hold that $F(\sigma) = 3$?
- Are there two different regular expressions σ, τ describing the same set such that $F(\sigma) \neq F(\tau)$?

Give short explanations for your answers.

Solution. $F(((\{00,11\}^* \cdot \emptyset) \cup \{00,11,22\})) = \max\{F((\{00,11\}^* \cdot \emptyset)), F(\{00,11,22\})\}$ $= \max\{F(\emptyset), F(\{00,11,22\})\} = \max\{0,2\} = 2$. In general, the function F of an expression σ for a set L is satisfies the following equation:

$$F(\sigma) = \begin{cases} 0 & \text{if } L = \emptyset; \\ 1 & \text{if } L = \{\varepsilon\}; \\ 2 & \text{if } L \text{ is finite and contains a nonempty word;} \\ 3 & \text{if } L \text{ is infinite.} \end{cases}$$

In particular, if σ, τ describe the same set then $F(\sigma) = F(\tau)$ and $F(\sigma) = 3$ iff σ describes an infinite set. One can verify above equation on F by induction: The conditions are hard-coded for lists of members of finite sets.

If $\sigma = (\tau \cup \rho)$ then $F(\sigma) = 0$ iff both $F(\tau), F(\rho) = 0$ iff both τ, ρ describe the empty set so that σ also describes the empty set; similarly σ describes $\{\varepsilon\}$ iff one of τ, ρ describes the set $\{\varepsilon\}$ and the other one either the same set or the empty set, so $F(\sigma) = 1$ iff $\max\{F(\tau), F(\rho)\} = 1$; $F(\sigma) = 2$ iff σ describes a finite set containing a non-empty string iff one of ρ, τ does and if both sets are finite iff $\max\{F(\rho), F(\tau)\} = 2$; $F(\sigma) = 3$ iff one of ρ, τ describe an infinite set iff at least one of $F(\rho), F(\tau)$ is 3.

Similarly one can verify the rules for $\sigma = \rho \cdot \tau$ with the special case in mind that the concatenation with an empty set gives the empty set.

Furthermore $F(\tau^*) = 3$ iff τ^* describes an infinite set iff τ contains a nonempty string iff $F(\tau) \geq 2$; $F(\tau^*) = 1$ iff τ^* describes the set $\{\varepsilon\}$ iff τ describes either $\emptyset$ or $\{\varepsilon\}$ iff $F(\tau) \leq 1$.

END OF PAPER