CS21004 Formal Languages and Automata Theory, Spring 2019–2020

Class Test 1

06–February–2020 | F116/F142, 06:45pm–07:45pm | Maximum marks: 20

Roll no: ____________ **Name:** ______________________________

[*Write your answers in the question paper itself. Be brief and precise. Answer all questions.*
If you use any example/result/formula covered in the class, just mention it, do not elaborate.]

1. The language $L_1 = \{uvv^r w \mid u, v, w \in \{a, b\}^+\}$ is regular. Here, v^r is the reverse of v.

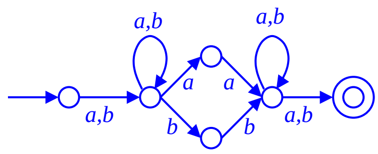
(a) Design a regular expression whose language is L_1. **(4)**

Solution L_1 consists precisely of all the strings over $\{a, b\}$ that contain the pattern aa or bb somewhere except at the two ends. Therefore a regular expression for L_1 can be

$$(a+b)(a+b)^*(aa+bb)(a+b)^*(a+b).$$

(b) Convert the regular expression of Part (a) to an equivalent NFA. **(4)**

Solution By inspection, we can convert the regular expression of Part (a) to the following NFA.



2. Let L be a regular language. Prove that $L' = \{w \in L \mid w^r \notin L\}$ is again regular (where w^r is the reverse of w). **(4)**

Solution Let L^r denote the reverse of L, that is, $L^r = \{w^r \mid w \in L\}$. We have $L' = L \cap \overline{L^r}$, where $\overline{L^r}$ is the complement of L^r (in Σ^*). Now, use the fact that regular languages are closed under reversal, complement, and intersection.

3. Prove/Disprove: If L is a non-regular language, then L^* must also be a non-regular language. **(4)**

Solution *False.* Take $L = \{a^{n!} \mid n \geqslant 1\}$. It is shown in the class that L is not regular. But since $a = a^{1!}$ is in L, we have $L^* = \{a^i \mid i \geqslant 0\} = \mathscr{L}(a^*)$ which is regular.

You may likewise take $L = \{a^p \mid p \text{ is a prime}\}$, note that $a^2, a^3 \in L$, and conclude that $L^* = \{\varepsilon\} \cup \{a^i \mid i \geqslant 2\} = \mathscr{L}(\varepsilon + aaa^*)$. $L = \{a^{2^n} \mid n \geqslant 0\}$ also works.

One can even take $\Sigma = \{a, b\}$, and $L = \{w \in \Sigma^* \mid \#a(w) \neq \#b(w)\}$. Since $a, b \in L$, it follows that $L^* = \Sigma^*$.

4. Let L be an infinite regular language accepted by a DFA with exactly k states. Prove that L must contain a string w such that $k \leqslant |w| < 2k$ (where $|w|$ is the length of w). **(4)**

Solution Since L is an infinite language, it cannot consist of strings of length $< k$ alone. Pick any string $w_0 \in L$ of length $|w_0| \geqslant k$. If $|w_0| < 2k$, we are done. So suppose that $|w_0| \geqslant 2k$. By the pumping out case of the pumping lemma applied on w_0, there exists a string $w_1 \in L$ such that $|w_1| < |w_0|$. Since $|w_0| \geqslant 2k$, and the pumped out string is of length at most k, we have $|w_1| \geqslant k$. If $|w_1| < 2k$, we are done. Otherwise, we apply the pumping lemma again on w_1 to get a string $w_2 \in L$ such that $k \leqslant |w_2| < |w_1|$. This process cannot be repeated infinitely often because the lengths of the strings $w_0, w_1, w_2, \ldots$ (each $\geqslant k$) are strictly decreasing. Therefore for some n, we must have $k \leqslant |w_n| < 2k$.

Use this space for *leftover answers* only. Do rough work in the extra sheet provided.