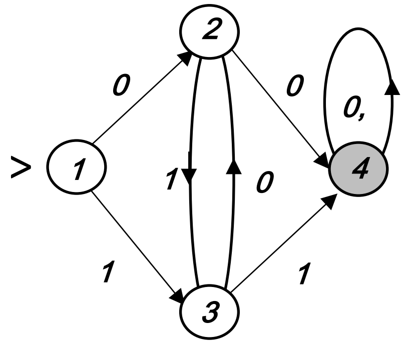
Homework #3 *due November 16, 2020, Monday before recitation*

(1) Consider the regular expression $\boldsymbol{E = (1+(0+101)^*)^*}$ **.**Draw an ***ε-NFA*** accepting the language corresponding to ***E*** above using as little number of states as possible ; compute and sketch the equivalent ***NFA*** without *ε*-transitions ; and finally compute the equivalent ***DFA*** accepting the language ***E*** above.

(2) Convert the following **DFA** to **RE** using the state elimination technique. Try to simplify the regular expression using the equivalence relations stated in class.



(3) For following languages, prove or disprove the statement that the language is ***regular***.

(a) $\boldsymbol{\{ww^R \mid w \in (0+1)^*\}}$ **,** where $\boldsymbol{w^R}$ stands for the string ***w*** written in reverse (backwards)

(b) *{****w*** *|* ***w*** *has same number of occurrences of* ***01*** *and* ***10*** *as substrings}*

(4) Consider the *Deterministic Finite Automata* ,

$\boldsymbol{A = (Q_A, \Sigma_A, \delta_A, q_{0A}, F_A)}$ and $\boldsymbol{B = (Q_B, \Sigma_B, \delta_B, q_{0B}, F_B)}$ with

$\boldsymbol{Q_A \cap Q_B = \phi}$ **,** $\boldsymbol{\Sigma_A \cap \Sigma_B = \phi}$ where $\boldsymbol{\phi}$ stands for the null set.

Let $\boldsymbol{L_A \subseteq \Sigma_A^*}$ and $\boldsymbol{L_B \subseteq \Sigma_B^*}$ be the languages accepted by ***A*** and ***B*** respectively and define the interleaved language:

$\boldsymbol{L_A \,||\, L_B := \{s\ \varepsilon\ (\Sigma_A \cup \Sigma_B)^* \mid s \uparrow_A \varepsilon\ L_A\ \ and\ s \uparrow_B \varepsilon\ L_B\}}$

where $\boldsymbol{s \uparrow_A}$ and $\boldsymbol{s \uparrow_B}$ stand for the projection of ***s*** on $\boldsymbol{\Sigma_A}$ and $\boldsymbol{\Sigma_B}$ obtained by erasing all the symbols of ***s*** in $\boldsymbol{\Sigma_B}$ and $\boldsymbol{\Sigma_A}$ respectively.

(a) Define the *interleaving product* ***A* || *B*** of ***A*** and ***B*** as a ***DFA*** that accepts the language $\boldsymbol{L_A \,||\, L_B}$

(b) Compute a ***DFA*** that accepts the language ***L = (0.1)***|| **(a.b)****

(5) Let ***A*** and ***B*** be ***DFA***s that accept languages $\boldsymbol{L_a}$ and $\boldsymbol{L_b}$ over an alphabet ***Σ***.

(a) Describe algorithm(s) using the product automaton $\boldsymbol{A \times B}$ to decide the following problems :

(i) $\boldsymbol{L_a = L_b}$;

(ii) $\boldsymbol{L_a}$ *strictly included in* $\boldsymbol{L_b}$;

(iii*)* $\boldsymbol{L_b}$ *strictly included in* $\boldsymbol{L_a}$*;*

(iv) *none of the first three!*

(b) State the complexity of your algorithm as an *O* function of ***n*** if the number of states of ***A*** and ***B*** are the same and are equal to ***n***.

(6) Problems from the main textbook

Exercise 4.1.2 ((b),(c),(h))

Exercise 4.3.3, 4.3.4

Exercises 4.4.2, 4.4.3