Zadanie 13. Skonstruuj automat skończony rozpoznający i wyrażenie regularne definiujące, nad alfabetem $\{a, b\}$, język słów, które nie zawierają wzorca *baba*.

Zadanie 15. Dodanie do definicji wyrażeń regularnych pozwolenia na użycie symbolu $\cap$, oznaczającego przekrój języków nie umożliwia reprezentowania nowych zbiorów, wyrażenia jednak stają się krótsze. Udowodnij, że użycie $\cap$ może wykładniczo skrócić wyrażenie.

Wskazówka: rozważyć język składający się z jednego słowa $(\ldots((a_0a_1)^2a_2)^2\ldots)^2$.

Zadanie 16. Czy istnieje wyrażenie regularne ϕ, oznaczające jakiś niepusty język regularny, takie że $L_{a\phi} = L_{\phi b}$? Czy istnieje wyrażenie regularne ϕ, oznaczające jakiś niepusty język regularny, takie że $L_{a^*\phi} = L_{\phi b^*}$?

4 Zadania o deterministycznych wyrażeniach regularnych.

Deterministic regular expressions, znane również jako *unambiguous regular expressions* pojawiły się kiedyś, jak się wydaje niechcący, w definicji standardu XML

Definicja. Niech ϕ będzie wyrażeniem regularnym nad alfabetem $\mathcal{A}$, a w słowem nad tym alfabetem. Niech f będzie funkcją, której argumentami są wystąpienia liter alfabetu w słowie w (czyli "kolejne litery słowa w"), a wartościami są wystąpienia liter w wyrażeniu ϕ. Powiemy, że f jest **poprawnym mapowaniem** w na ϕ, jeśli zachodzi któryś z warunków:

1. ϕ jest słowem nad $\mathcal{A}$, $\phi = w$ i f jest identycznością lub $\phi = \varepsilon$ i w jest puste;
2. $\phi = \phi_1 + \phi_2$ i f jest poprawnym mapowaniem w na ϕ_1 lub f jest poprawnym mapowaniem w na ϕ_2;
3. $\phi = \phi_1\phi_2$, $w = w_1w_2$ i f ograniczona do w_1 jest poprawnym mapowaniem tego słowa na ϕ_1, zaś f ograniczona do w_2 jest poprawnym mapowaniem tego słowa na ϕ_2;
4. $\phi = \psi^*$, $w = w_1w_2\ldots w_k$, dla jakiegoś $k \geqslant 0$ i dla każdego $1 \leqslant i \leqslant k$ funkcja f ograniczona do w_i jest poprawnym mapowaniem w_i na ψ.

Intuicja jest taka, że poprawne mapowanie słowa przyporządkowuje każdej jego literze, literę wyrażenia z której ta litera słowa "się wzięła". Wyrażenie ϕ jest **deterministycznym wyrażeniem regularnym**, jeśli dla każdego $w \in L_\phi$ istnieje dokładnie jedno poprawne mapowanie w na ϕ. Deterministyczne wyrażenie regularne pozwala odczytać, które litery w słowie biorą się z których liter w wyrażeniu, ale to odczytanie następuje dopiero, gdy znamy całe słowo. inaczej jest dla detereministycznych on-line wyrażeń regularnych. Wyrażenie regularne ϕ jest **deterministyczne on-line**, jeśli dla każdych słów ww_1, $ww_2 \in L_\phi$ i każdych funkcji f_1, f_2, będących poprawnymi mapowaniami słów (odpowiednio) ww_1 i ww_2 na ϕ, funkcje f_1 i f_2 zgadzają się na prefiksie w.

Zadanie 17. **a.** Które z poniższych wyrażeń są deterministyczne, a które są deterministyczne on-line?
i. $0^*10^* + 0^*$
ii.$(0+1)^*1(0+1)$
iii.$(0+1)(0+2)^* + (1+2)(0+1)^* + (0+2)(1+2)^*$
b. Znajdź deterministyczne wyrażenie regularne oznaczające język tych wszystkich słów nad alfabetem zerojedynkowym, które zawierają wzorzec 101.

Zadanie 18. Czy dla każdego języka regularnego istnieje deterministyczne on-line wyrażenie regularne, które go definiuje?

Zadanie 19. Znajdź deterministyczne on-line wyrażenie regularne oznaczające język tych wszystkich słów nad alfabetem zerojedynkowym, które zawierają jedną lub dwie jedynki.

Zadanie 23. Załóżmy, że L jest pewnym językiem regularnym. Czy język $L/2 = \{w : \exists v\ vw \in L\ \wedge |v| = |w|\}$ jest regularny?

• Transducer Moore'a to krotka $\langle \Sigma, \Sigma_1, Q, q_0, \delta, \sigma \rangle$ gdzie $\langle \Sigma, Q, q_0, \emptyset, \delta \rangle$ jest DFA (z pustym zbiorem stanów akceptujących) i gdzie $\sigma : Q \to \Sigma_1^*$ dla pewnego alfabetu Σ_1. Jeśli $T = \langle \Sigma, \Sigma_1, Q, q_0, \delta, \sigma \rangle$ jest transducerem Moore'a to $f_T : \Sigma^* \to \Sigma_1^*$ jest zdefiniowana jako $f_T(\varepsilon) = \varepsilon$ oraz $f_T(wa) = (f_T(w))\sigma(\hat{\delta}(wa, q_0))$.

• Transducer Mealy'ego zdefiniowany jest analogicznie, z tą różnicą że $\sigma : Q \times \Sigma \to \Sigma_1^*$ oraz $f_T(wa) = (f_T(w))\sigma(\hat{\delta}(w, q_0), a)$.

• Transducery T i T' są *równoważne* jeśli funkcje f_T i $f_{T'}$ są równe.

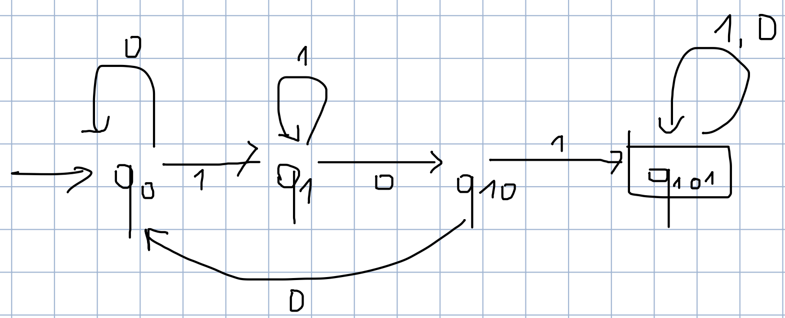
• Dla języków $A \subseteq \Sigma^*$ i $B \subseteq \Sigma_1^*$ definiujemy $A \leqslant_{reg} B$ jeśli istnieje transducer T (Moore'a lub Mealy'ego) taki że dla kazdego $w \in \Sigma^*$ zachodzi $w \in A$ w.t.w. gdy $f_T(w) \in B$.

Zadanie 53. Pokaż że dla każdego transducera Moore'a istnieje równoważny transducer Mealy'ego. Pokaż że dla każdego transducera Mealy'ego istnieje równoważny transducer Moore'a.

Zadanie 54. Pokaż że jeśli $A \leqslant_{reg} B$ i B jest regularny to A też.

Zadanie 55. Pokaż że dla każdego n istnieje transducer Mealy'ego $T = \langle \Sigma, \Sigma_1, Q, q_0, \delta, \sigma \rangle$ taki że $|Q| = |\Sigma| = n$ i że każdy transducer Moore'a równoważny T ma przynajmniej n^2 stanów.

Zadanie 56. Niech $A \subseteq \{(,), [,], \langle, \rangle\}^*$ będzie językiem poprawnie rozstawionych nawiasów trzech rodzajów zaś $B \subseteq \{(,), [,]\}^*$ językiem poprawnie rozstawionych nawiasów dwóch rodzajów. Pokaż że $A \leqslant_{reg} B$. *Wskazówka: każde słowo produkowane przez σ ma się składać z dwóch symboli.*



(1+(eps+0+11*00)(0+11*00)*1) 1*0 (1+1(1+0)*(eps+0+1))

$q_0 = q_0 0 + q_{10} 0$

$q_1 = q_0 1 + q_1 1$

$q_{10} = q_1 0$

$\underline{q_{101} = q_{101} (1+0) + q_{10} 1}$

$q_{101} = q_{10} 1(1+0)^*$

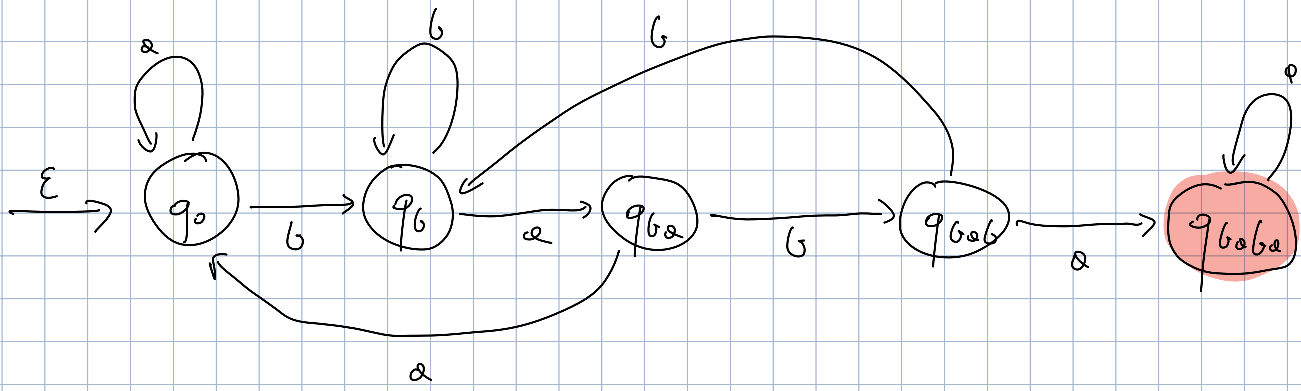
$q_1 = q_0 1(1)^*$

$q_{10} = q_0 1(1)^* 0$

$q_0 = (0 + 1(1)^* 00)^*$

$(0+1(1)^*00)^* 1(1)^* 0\, 1(1+0)^*$

Zadanie 13. Skonstruuj automat skończony rozpoznający i wyrażenie regularne definiujące, nad alfabetem $\{a, b\}$, język słów, które nie zawierają wzorca *baba*.



$$q_0 = \varepsilon + q_0 a + q_{ba} a$$
$$q_b = q_0 b + q_b b + q_{bab} b$$
$$q_{ba} = q_b a$$
$$q_{bab} = q_{ba} b \Rightarrow q_{bab} = q_b ab$$

$$q_b = q_0 b + q_b b + q_b abb$$
$$q_b = q_0 b + q_b (b + abb)$$
$$q_b = q_0 b (b + abb)^*$$

$$q_0 = \varepsilon + q_0 a + q_b aa$$
$$q_0 = \varepsilon + q_0 a + q_0 b (b + abb)^* aa$$
$$q_0 = (a + b(b + abb)^* aa)^*$$

$$q_0 = (a + b(b + abb)^* aa)^*$$
$$q_b = (a + b(b + abb)^* aa)^* b (b + abb)^*$$
$$q_{ba} = (a + b(b + abb)^* aa)^* b (b + abb)^* a$$
$$q_{bab} = (a + b(b + abb)^* aa)^* b (b + abb)^* ab$$

Ostatecznie $\varphi = (a + b(b + abb)^* aa)^* + (a + b(b + abb)^* aa)^* b (b + abb)^*$
$+ (a + b(b + abb)^* aa)^* b (b + abb)^* a$
$+ (a + b(b + abb)^* aa)^* b (b + abb)^* ab$

$\varphi_{1,1}^{5} = (a + bb^*aa + (bab)^* baa)^*$

$\varphi_{1,2}^{5} = a^*b(aaa^*b + abb + b)^*$

$\varphi_{1,3}^{5} = a^*bb^*a(aa^*bb^*a + bbb^*a)^*$

$\varphi_{1,4}^{5} = a^*bb^*ab($

$\varphi_{1,4}^{4} = a^*bb^*a(aa^*bb^*a)^*b$

$\varphi_{4,4}^{4} = bb^*a(aa^*bb^*a)^*b$

$\varphi_{1,4}^{5} = a^*bb^*a(aa^*bb^*a)^*bbb^*a(aa^*bb^*a)^*b$

Zadanie 16. Czy istnieje wyrażenie regularne ϕ, oznaczające jakiś niepusty język regularny, takie że $L_{a\phi} = L_{\phi b}$? Czy istnieje wyrażenie regularne ϕ, oznaczające jakiś niepusty język regularny, takie że $L_{a^*\phi} = L_{\phi b^*}$?

a) Nie.

Załóżmy, że istnieje takie φ.

Weźmy $a^k w \in L_{a\varphi}$, k - najmniejsze

Wtedy $a^{k-1} w \in L_\varphi$, czyli $a^{k-1} wb \in L_{\varphi b}$

Czyli $a^{k-1} wb \in L_{a\varphi}$, co jest sprzeczne z minimalnością k ↯

b) Niech $\varphi = a^*b^*$

Wtedy $L_{a^*\varphi} = L_{\varphi b^*}$.

Zadanie 15. Dodanie do definicji wyrażeń regularnych pozwolenia na użycie symbolu $\cap$, oznaczającego przekrój języków nie umożliwia reprezentowania nowych zbiorów, wyrażenia jednak stają się krótsze. Udowodnij, że użycie $\cap$ może wykładniczo skrócić wyrażenie.

Wskazówka: rozważyć język składający się z jednego słowa $(\ldots((a_0a_1)^2a_2)^2\ldots)^2$.

Niech $w_k = (\ldots(a_0a_1)^2a_2)^2\ldots)a_k)^2 = (w_{k-1}a_k)^2$

- Zauważmy, że w wyrażeniu regularnym opisującym $L_k = \{w_k\}$ nie mogą wystąpić znaki $+$ i $*$, bo w innym przypadku otrzymamy większy język. Zatem długość słowa w_k będzie długością wyr. reg. opisującego L_k.

$|w_k| = |w_{k-1}a_kw_{k-1}a_k| = 2|w_{k-1}|+2$

$|w_k| = 3\cdot 2^k - 2$

- Rozważmy teraz $\varphi_k = a_0 + a_1 + \ldots + a_k$

Wtedy φ_k^* to będą wszystkie słowa niezawierające $a_{k+1}, a_{k+2}, \ldots$

Możemy teraz zapisać wyr. reg. opisujące L_k:

$\psi_k = (\psi_{k-1}a_k)^* \cap (\varphi_{k-1}^*a_k\varphi_{k-1}^*a_k)$ oraz $\psi_1 = w_1$

Pokażemy, że $L_{\psi_k} = L_k$ indukcyjnie:

dla $k=1$ ok

załóżmy, że dla k działa, wtedy dla $k+1$ też:

$w_{k+1} = w_ka_{k+1}w_ka_{k+1}$

zatem spełnia ono $(\psi_ka_k)^i$ oraz $(\varphi_k^*a_{k+1}\varphi_k^*a_{k+1})$, bo a_{k+1} występuje dokładnie 2 razy
dla $i=2$

Pokażemy teraz długość wyrażenia

$|\psi_k| = |\psi_{k-1}| + 3|a_k| + 2|\varphi_{k-1}| + 4$

$|\psi_k| = |\psi_{k-1}| + 3 + 2(k+1+k) + 4$

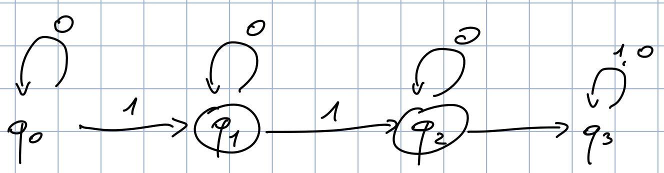
$|\psi_k| = |\psi_{k-1}| + 4k + 9$

$|\psi_k| - |\psi_{k-1}| = O(k)$

$|\psi_k| = \sum_{n=1}^{k}(|\psi_k| - |\psi_{k-1}|) + |\psi_1| = O(k^2) + 4 = O(k^2)$

Zadanie 19. Znajdź deterministyczne on-line wyrażenie regularne oznaczające język tych wszystkich słów nad alfabetem zerojedynkowym, które zawierają jedną lub dwie jedynki.

$\varphi = 0^*10^*(\varepsilon + 10^*)$



Niech $ww_1, ww_2 \in L_\varphi$

$w = a_1 \dots a_n$, $w_1 = b_1 \dots b_m$, $w_2 = c_1 \dots c_k$

Rozważmy przypadki:

1) ww_1, ww_2 zawierają jedną jedynkę

a) ta jedynka jest w w_i:

wtedy $w_1 \dots w_{i-1} \to 0_1$

$w_i \to 1_1$

$w_{i+1}, \dots, w_n \to 0_2$

czyli się zgadza

b) jedynka jest za w:

wtedy $w = 0^n$, czyli $w_1, \dots, w_n \to 0_1$ ok

2) bso ww_1 ma jedną jedynkę, ww_2 ma dwie

a) w ma jedną jedynkę: j.w.

b) w nie ma jedynek: j.w.

3) dwa mają dwie jedynki:

a) w nie ma jedynek: j.w.

b) w ma jedną jedynkę: j.w

c) w ma dwie jedynki: w_i, w_j $i<j$

wtedy $w_1, \dots, w_{i-1} \to 0_1$, $w_i \to 1_1$, $w_{i+1}, \dots, w_{j-1} \to 0_2$, $w_j \to 1_2$, $w_{j+1}, \dots, w_n \to 0_3$ ok.

• Transducer Moore'a to krotka $\langle \Sigma, \Sigma_1, Q, q_0, \delta, \sigma \rangle$ gdzie $\langle \Sigma, Q, q_0, \emptyset, \delta \rangle$ jest DFA (z pustym zbiorem stanów akceptujących) i gdzie $\sigma : Q \to \Sigma_1^*$ dla pewnego alfabetu Σ_1. Jeśli $T = \langle \Sigma, \Sigma_1, Q, q_0, \delta, \sigma \rangle$ jest transducerem Moore'a to $f_T : \Sigma^* \to \Sigma_1^*$ jest zdefiniowana jako $f_T(\varepsilon) = \varepsilon$ oraz $f_T(wa) = (f_T(w))\sigma(\hat{\delta}(wa, q_0))$.

• Transducer Mealy'ego zdefiniowany jest analogicznie, z tą różnicą że $\sigma : Q \times \Sigma \to \Sigma_1^*$ oraz $f_T(wa) = (f_T(w))\sigma(\hat{\delta}(w, q_0), a)$.

• Transducery T i T' są *równoważne* jeśli funkcje f_T i $f_{T'}$ są równe.

• Dla języków $A \subseteq \Sigma^*$ i $B \subseteq \Sigma_1^*$ definiujemy $A \leqslant_{reg} B$ jeśli istnieje transducer T (Moore'a lub Mealy'ego) taki że dla kazdego $w \in \Sigma^*$ zachodzi $w \in A$ w.t.w. gdy $f_T(w) \in B$.

Zadanie 53. Pokaż że dla każdego transducera Moore'a istnieje równoważny transducer Mealy'ego. Pokaż że dla każdego transducera Mealy'ego istnieje równoważny transducer Moore'a.

Weźmy dowolny transducer Moore'a $T = \langle \Sigma, \Sigma_1, Q, q_0, \delta, \sigma \rangle$

Skonstruujemy równoważny transducer Mealy'ego $T' = \langle \Sigma, \Sigma_1, Q, q_0, \delta, \sigma' \rangle$

Niech $\sigma' : Q \times \Sigma \to \Sigma_1^*$, $\sigma'(q, a) = \sigma(\delta(a, q))$

Indukcyjnie pokażemy $f_T(w) = f_{T'}(w)$:

1) $f_T(\varepsilon) = \varepsilon = f_{T'}(w)$ ok.

2) załóżmy, że $f_T(w) = f_{T'}(w)$.

pokażemy $\forall_{a \in \Sigma}\ f_T(wa) = f_{T'}(wa)$

$f_T(wa) = ((f_T(w))\sigma(\hat{\delta}(wa, q_0)) \overset{zał.}{=} ((f_{T'}(w))\sigma(\hat{\delta}(wa, q_0)) =$

$= ((f_{T'}(w))\sigma(\delta(a, \hat{\delta}(w, q_0)))$

$= (f_{T'}(w))\sigma'(\hat{\delta}(w, q_0), a) = f_{T'}(wa)$ ok.

Zatem T i T' są równoważne $\square$

Teraz w drugą stronę $\underset{\text{Mealy'ego}}{T = \langle \Sigma, \Sigma_1, Q, q_0, \delta, \sigma \rangle}$, $\underset{\text{Moore'a}}{T' = \langle \Sigma', \Sigma_1, Q', q_0', \delta', \sigma' \rangle}$

$q_0' = (\square, q_0)$

Niech $\Sigma' = \Sigma \cup \{\square\}$ $Q' = Q \times \Sigma'$, przyjmujemy $\delta(q, \square) = q$

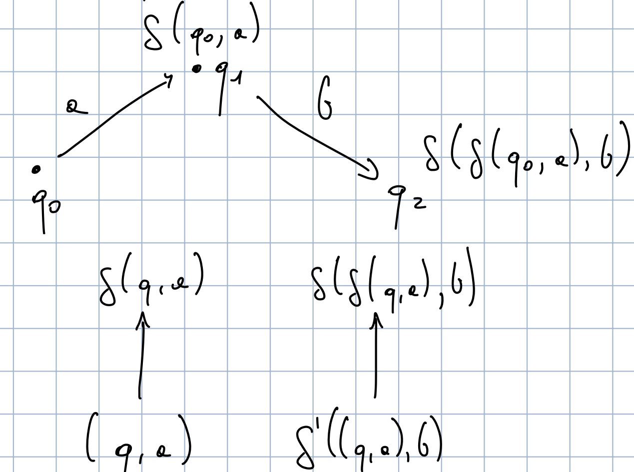
Niech $\delta'((q, a), b) = (\delta(q, a), b)$

Oraz $\sigma'((q, a)) = \sigma(q, a)$

Teraz pokażemy $f_T(w) = f_{T'}(w)$:

$f_{T'}(wa) = (f_{T'}(w))\,\sigma'(\hat{\delta}'(q_0', wa))$

$\langle \Sigma, Q, q_0, \delta, \emptyset\rangle$:



$\sigma'(\hat{\delta}'(q_0', wa)) =$

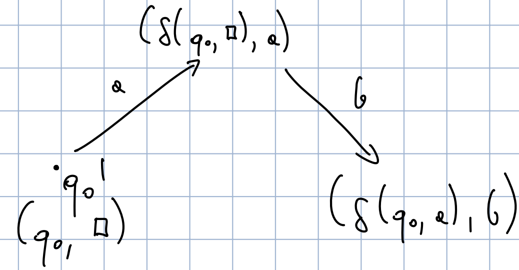
$= \sigma'(\delta'(\hat{\delta}'(q_0', w), a))$

$= \sigma'((\delta(\hat{\delta}'(q_0', w)), a))$

$= \sigma(\delta(\hat{\delta}'(q_0', w)), a)$

$= \sigma(\hat{\delta}(q_0, w), a)$ $\square$

$\langle \Sigma', Q', q_0', \delta', \emptyset\rangle$:



Zadanie 54. Pokaż że jeśli $A \leqslant_{reg} B$ i B jest regularny to A też.

$A \subseteq \Sigma^*$, $B \subseteq \Sigma_1^*$

Skoro $A \leqslant_{reg} B$, to istnieje transducer Mealy'ego $T = \langle \Sigma, \Sigma_1, Q_T, q_T, \delta_T, \sigma\rangle$

t. że $\forall_{w \in \Sigma^*}\ w \in A \iff f_T(w) \in B$

Niech $\mathcal{B}$ – DFA rozpoznający B

$= \langle \Sigma_1, Q_B, q_B, \delta_B, F_B\rangle$

Skonstruujemy DFA $\mathcal{A} = \langle \Sigma, Q_A, q_A, \delta_A, F_A\rangle$ rozpoznający A

Niech $Q_A = Q_T \times Q_B$

$q_A = \langle q_T, q_B\rangle$

$F_A = \{\langle q, p\rangle : p \in F_B\}$

$\delta_A(q', a) = \delta_A(\langle q, p\rangle, a) = \langle \delta_T(q, a), \hat{\delta}_B(p, \sigma(q, a))\rangle$

Pokażemy teraz $\hat{\delta}_A(q_A, w) = \langle \hat{\delta}_T(q_T, w), \hat{\delta}_B(q_B, f_T(w)) \rangle$

dla $w = \varepsilon$ $\hat{\delta}_A(q_A, w) = \langle q_T, q_B \rangle = \langle \hat{\delta}_T(q_T, \varepsilon), \hat{\delta}_B(q_B, \varepsilon) \rangle$ ok

załóżmy, że $\hat{\delta}_A(q_A, w) = \langle \hat{\delta}_T(q_T, w), \hat{\delta}_B(q_B, f_T(w)) \rangle$

pokażemy $\forall_{a \in \Sigma}$ $\hat{\delta}_A(q_A, wa) = \langle \hat{\delta}_T(q_T, wa), \hat{\delta}_B(q_B, f_T(wa)) \rangle$:

$$\hat{\delta}_A(q_A, wa) = \delta_A(\hat{\delta}_A(q_A, w), a) = \delta_A(\langle \hat{\delta}_T(q_T, w), \hat{\delta}_B(q_B, f_T(w)) \rangle, a)$$
$$\overset{\text{zał.}}{=} \langle \delta_T(\hat{\delta}_T(q_T, w), a), \hat{\delta}_B(\hat{\delta}_B(q_B, f_T(w)), \sigma(\hat{\delta}_T(q_T, w), a)) \rangle$$
$$= \langle \hat{\delta}_T(q_T, wa), \hat{\delta}_B(q_B, (f_T(w)\sigma(\hat{\delta}_T(q_T, w), a)) \rangle$$
$$\overset{\text{def.}}{=} \langle \hat{\delta}_T(q_T, wa), \hat{\delta}_B(q_B, f_T(wa)) \rangle$$

Zatem

$w \in L_{\mathcal{A}} \iff \hat{\delta}_A(q_A, w) \in F_A \iff \hat{\delta}_B(q_B, f_T(w)) \in F_B \iff f_T(w) \in B \iff w \in A_B$

Zadanie 55. Pokaż że dla każdego n istnieje transducer Mealy'ego $T = \langle \Sigma, \Sigma_1, Q, q_0, \delta, \sigma \rangle$ taki że $|Q| = |\Sigma| = n$ i że każdy transducer Moore'a równoważny T ma przynajmniej n^2 stanów.

Niech $T = \langle \Sigma, \Sigma_1, Q, q_0, \delta, \sigma \rangle$, $|Q| = |\Sigma| = n$

T – transducer Mealy'ego

Zał. nie wprost, że $T' = \langle \Sigma, \Sigma_1, Q', q_0', \delta', \sigma' \rangle$ – transducer Moore'a równoważny T.

$|Q'| < n^2$

Niech $\Sigma = \{a_0, \dots, a_{n-1}\}$ $Q = \{q_0, \dots, q_{n-1}\}$

Oraz $\Sigma_1 = \{b_0, \dots, b_{n^2-1}\}$

Niech $k = |Q'|$

Ustalmy $\sigma(q_i, a_j) = b_{ni+j}$, $\delta(q_i, a_j) = q_{(i+j) \bmod n}$

Wtedy:

- $f_T(a_i a_j) = f_T(a_i)\, \sigma(\delta(q_0, a_i), a_j)$
$= \sigma(\delta(q_0, \varepsilon), a_i)\, \sigma(\delta(q_0, a_i), a_j)$

$$= \sigma(q_0, a_i)\,\sigma(q_i, a_j)$$
$$= b_i\, b_{ni+j}$$

- $\sigma' : Q' \to \Sigma_1^*$

zatem $|\sigma'[Q']| \leqslant k < n^2$,

czyli w szczególności mamy co najwyżej k ostatnich liter

Zatem istnieje $b \in \Sigma_1$, które nie jest ostatnią literą każdego $\sigma'[q'] : q' \in Q'$

Niech $b = b_{ni+j}$ dla pewnych $0 \leqslant i,j \leqslant n-1$

Wtedy $f_T(a_i a_j) = b_i b_{ni+j}$, czyli zostanie wyprodukowane,

ale $f_{T'}(a_i a_j) = f_T(a_i a_j)$ co jest sprzeczne z obserwacją

Zadanie 56. Niech $A \subseteq \{(,),[,],\langle,\rangle\}^*$ będzie językiem poprawnie rozstawionych nawiasów trzech rodzajów zaś $B \subseteq \{(,),[,]\}^*$ językiem poprawnie rozstawionych nawiasów dwóch rodzajów. Pokaż że $A \leqslant_{reg} B$. *Wskazówka: każde słowo produkowane przez σ ma się składać z dwóch symboli.*

Niech $\Sigma_A = \{(,),[,],\langle,\rangle\}$

$\Sigma_B = \{(,),[,]\}$

$T = \langle \Sigma_A, \Sigma_B, Q, q_0, \delta, \sigma \rangle$ - transducer Mealy'ego

Niech $Q = \{q_0\}$

Ustalmy σ:

$\sigma(q_0, \langle) = ([$ $\quad$ $\sigma(q_0, \rangle) =])$

$\sigma(q_0, [) = ([[$ $\quad$ $\sigma(q_0,]) =]])$

$\sigma(q_0, '(') = ([[[$ $\quad$ $\sigma(q_0, ')') =]]])$

Lepsze kodowanie:

$(\to (($ $\quad$ $) \to))$

$[\to ([$ $\quad$ $] \to])$

$\langle \to [($ $\quad$ $\rangle \to)]$

- $w \in A \Rightarrow f_T(w) \in B$

  wynika z def. σ

- $w_T(w) \in B \Rightarrow w \in A$

  $w_T(w)$ jest poprawnym nawiasowaniem

  zauważmy, że w każdej wartości σ

  słowo zaczyna się od (, a kończy na [

  ((—)) —], —)) —)

  zatem jednoznacznie możemy zrekonstruować w

  tym samym dzięki def. σ widzimy, że $w \in A$

Zadanie 0

Język nieskończonych słów, które zawierają sk. wiele zer

nie daje się rozstrzygnąć deterministycznym automatem Büchiego

Niech $\Sigma = \{0,1\}$

$A = \langle \Sigma, Q, q_0, \delta, F \rangle$ - deterministyczny automat Büchiego rozstrzygający L

Niech $|Q| = n$

Wiemy, że 1^ω jest rozpoznawany przez A

Niech i_1 - minimalny t.że $\hat{\delta}(q_0, 1^{i_1}) \in F$

Wiemy, że $1^{i_1} 0 1^\omega$ jest rozpoznawany przez A

Niech i_2 - minimalny t.że $\hat{\delta}(q_0, 1^{i_1} 0 1^{i_2}) \in F$

Kontynuując otrzymujemy ciąg indeksów i_n

t.że $1^{i_1} 0 1^{i_2} 0 1^{i_3} 0 \ldots$ jest rozpoznawany przez A,

bo odwiedzimy F niesk. wiele razy.

Ale to słowo ma niesk. wiele zer $\lightning$

Zadanie 18. Czy dla każdego języka regularnego istnieje deterministyczne on-line wyrażenie regularne, które go definiuje?

Nie dla każdego:

$(0+1)^* 1 (1+0)$ $\ddot{-}$

0^* $\quad L=\{w \in 0^* : |w|=2n \vee |w|=3n\}$

$[00(\varepsilon+0)]^*$ $\quad \ddot{\smile}?$

L1

φ, ψ - regex

$|L_\varphi|>0,\ L_\psi \neq \emptyset,\ L_\psi \neq \{\varepsilon\}$

$\varphi\psi$ nie jest d.o.

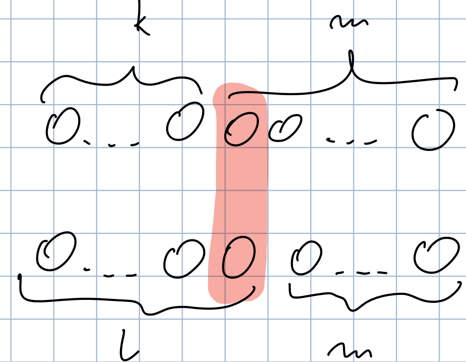
istnieje k, l, m takie że

$0^k, 0^l \in \varphi$

$0^m \in \psi$

$m>0$

bso $l>k$



Jeśli $|L_\varphi|>1$, to φ^* nie jest d.o.

$L_\varphi = \{0^k, 0^l\}$

Wtedy $0^k + 0^l$ można uzyskać na dwa sposoby

Czyli + nie może być pod gwiazdką

$(0^t)^* \varphi$ – nie działa

$(0^t)^* + \varphi$

$\varphi + \varphi$ i w dr jest niepuste słowo,
to już nie jest d.o.

$\boxed{0}0$ $\boxed{0}00$