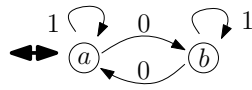


Cvičení z automatů a gramatik - 3

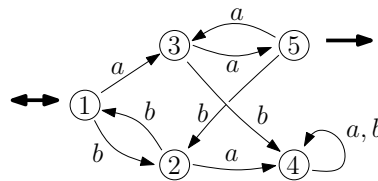
7. a 8. března 2018

Probrané příklady

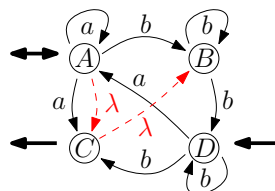
- Homomorfismy automatů: definice, zachování přijímaného jazyka.
 - Řekneme, že A je *homomorfní na* B , pokud existuje (automatový) homomorfismus $h : Q_A \rightarrow Q_B$. Je tato relace reflexivní, symetrická, tranzitivní ?
 - Uvažme následující konečný automat C . Nalezněte konečné automaty A, B homomorfní na C (a přitom neizomorfní s C) takové, že A není homomorfní na B a zároveň B není homomorfní na A .



- Automatová kongruence, podílový automat (faktorstruktura).
 - Je stavová ekvivalence po i krocích automatovou kongruencí?
 - Uveďte příklad automatové kongruence jiný než je stavová ekvivalence.
- Redukt: jednoznačnost. Minimalizujte následující konečný automat.



- Nedeterministický konečný automat: formální definice, výpočet, přijímaný jazyk, interpretace pomocí větvících výpočtů a pomocí uhodnutí přijímacího výpočtu.
 - Může být množina počátečních stavů prázdná?
 - Můžeme přidat podmínku, že z každého stavu je pro každé písmeno definován aspoň jeden přechod?
 - Sestrojte deterministický a nedeterministický automat (s co nejmenší množinou stavů) rozpoznávající jazyk $L = \{w \in \{a, b\}^*; w \text{ končí na } abba\}$.
- Nedeterministické konečné automaty přijímají právě regulární jazyky.
 - Převedte následující nedeterministický automat (bez λ -přechodů) na deterministický (množinovou konstrukcí).
 - Je výsledný deterministický automat (vždy) redukováný?



6. Rozpoznávání doplňku nedeterministickými konečnými automaty.
- (a) Jaký jazyk dostaneme po přehození koncových/nekoncových stavů u deterministického automatu?
 - (b) A u nedeterministického automatu?
 - (c) Jak můžeme zadefinovat nový typ nedeterministických konečných automatů (přesněji nový typ přijímaného jazyka), abychom po přehození typu a koncových/nekoncových stavů dostali doplněk původního jazyka?

Domácí úkol

7. Pro každé $n \geq 1$ nalezněte jazyk L_n (nad libovolnou abecedou) takový, že každý deterministický automat rozpoznávající L_n má aspoň 2^n stavů, a zároveň L_n lze rozpoznat nedeterministickým automatem s n stavy.

Automaty a gramatiky, DÚ3

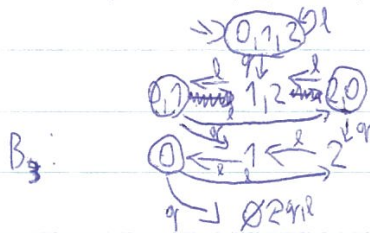
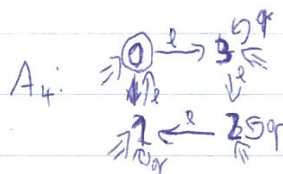
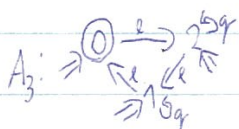
Vít Fojtík 16

Místo jazyka budu uvažovat rovnou o příslušných automatech. Díky větě, že pokud jsou dva DFA ekvivalentní, jsou jejich redukty isomorfní, stačí k příslušnému NFA najít ekvivalentní DFA, který je redukovaný, protože potom je sám svým reduktem, tedy je isomorfní reduktu lib. ekvivalentního DFA, tedy nemá větší počet stavů.

Pro $m \geq 1$ def. NFA A_m takto: $A_m = \langle m, \{1, q\}, \delta_m, m, \{0\} \rangle$, kde $\forall k \in m (\delta_m(k, 1) = \{k-1 \pmod m\})$, $\forall k \in m (\delta_m(k, q) = \{k\})$, $\delta_m(0, q) = \emptyset$. (značení: $l(k) = \delta_m(k, 1)$, $q(k) = \delta_m(k, q)$). (Moje úmysly se snad objasní dále). Jako hledaný jazyk def. $L_m = L(A_m)$. Nyní pro A_m sestrojím potenční DFA B_m (jehož ekvivalence plyne z konstrukce, resp. cvičení) a ukážu, že je redukovaný.

$B_m = \langle P(m), \{1, q\}, \gamma_m, m, \{x \in P(m) \mid 0 \in x\} \rangle$, $\gamma_m(x, 1) = \cup_{i \in x} i$, $\gamma_m(x, q) = \emptyset$ (opět budu značit $l(x)$,

$q(x)$, zmatení snad nehrozí). Intuice: $x \in P(m)$ si představuji jako označení určitých pozic na "pásce" $0, \dots, m-1$, kde na začátku jsou označeny všechny pozice, l zrotuje pozice doleva a q vynechá pozici 0. Např.:



Dosažitelnost všech stavů: "klesavou indukcí" dle $|x|$, $x \in P(m)$: $|x|=m \Rightarrow x=m \Rightarrow x$ je počáteční stav.

Bud' $|x|=2-1$ a j nejmenší takové, že $j \notin x$. Def. $x' = x \cup \{j\}$. Potom $l^{m-j}(q(l^0(x))) = x$ (Bud' $x' = m$, pokud $m \neq j$, pak $l^0(m) = \{0\}$, $q(l^0(m)) = \{m\}$. Pokud $m=j$, pak $q(l^0(m)) = \emptyset$).

Redukovanost: Pro spor mějme x, x' ekvivalentní stavy, $x \neq x'$. Bud' z nejmenší $z \in x$ a $z \notin x'$.

Potom $l^z(x) \in F_{B_m}$ a $l^z(x') \notin F_{B_m}$ (Indukcí: $z=0$, $y \in F_{B_m} \Leftrightarrow 0 \in y$, $0 \in x$ a $0 \notin x' \Rightarrow x \in F_{B_m}$ a $x' \notin F_{B_m}$. Pro $z > 0$:

$l(x)$ je ekv. $l(x')$ z def. stavové ekvivalence, navíc $\forall j < z (j \in x \Leftrightarrow j \in x')$ a $z \in l(x)$, $z \notin l(x')$, takže jsou splněny předpoklady a z IP je $l^z(l(x)) \in F_{B_m}$ a $l^z(l(x')) \notin F_{B_m}$.) $\sim \zeta$

Tím je snad hotovo.