
PAL: 9. cvičení

24. 11. 2022

Př. 7/11: regulární soupeři

Rozhodněte, zda regulární výrazy $(01 + 0) * 0$ a $0(10 + 0)^*$ popisují stejný regulární jazyk.

Př. 7/6: automat z regulárního výrazu

Sestavte automat, který v textu nad abecedou $\{a, b, c\}$ vyhledává všechna slova popsaná regulárním výrazem $(ac^* + bb)^* a$.

Př. 7/7: uspořádaná slova

Mějme abecedu $A = \{a, b, c, \dots, z\}$. Pořadové číslo znaku a bude 1, pořadové číslo znaku b bude 2, atd., až pořadové číslo znaku z bude 26. Slovo nad A nazveme uspořádané, pokud pro každý jeho znak platí, že všechny znaky za ním ve slově následující mají vyšší pořadové číslo než tento znak. Sestavte NKA, který vyhledá v textu nad abecedou A všechna uspořádaná slova.

Př. 7/8: řetězce se stejným počtem znaků

Sestavte NKA nad abecedou $\{0, 1, 2\}$, který v textu vyhledá všechny řetězce obsahující stejný počet znaků 0, 1 i 2.

Operace nad jazyky. Přibližné
vyhledávání v textu pomocí
konečných automatů

Př. 8/0: skládání automatů

Ze dvou NKA A_1 a A_2 přijímajících jazyky L_1 resp. L_2 vytvořte dva NKA, které budou přijímat jazyk $L_1 \cup L_2$ resp. $L_1 \cap L_2$. (*Doporučení: Stavů A_1 si označte písmeny, stavy A_2 čísly, aby bylo skládání přehledné.*)

a) $L_1 = \{0\}, L_2 = \{1\},$

b) $L_1 = \{(00)^*\}, L_2 = \{(000)^*\}.$

Př. 8/1*: skládání automatů 2

Nad abecedou $\{0, 1\}$, jsou dány dva jazyky L_1 a L_2 . Slova L_1 jsou popsána výrazem $0^* 1^* 0^* 1^* 0^*$, slova L_2 jsou popsána výrazem $(01 + 10)^*$.

Sestrojte konečné automaty rozpoznávající jazyk:

- a) $L_1 \cup L_2$,
- b) $L_1 \cap L_2$.

Př. 8/2: konečný průnik

Automat A_1 rozpoznává jazyk L_1 , automat A_2 rozpoznává jazyk L_2 . Oba automaty mají n stavů. Abeceda pro oba jazyky je shodná a má k znaků. Jaká je asymptotická složitost algoritmu, který efektivně určí, zda jazyk $L_1 \cap L_2$ je konečný?

Př. 8/9: Hamming a Levenshtein

Napište všechna slova, která mají od vzorku aba nad abecedou $\{a, b, c\}$ Hammingovu resp. Levenshteinovu vzdálenost rovnu 1.

Př. 8/8: Hamming vs. Levenshtein

Označme symbolem $HD(v, w)$ Hammingovu vzdálenost slov v a w nad abecedou A , symbolem $LD(v, w)$ Levenshteinovu vzdálenost těchto slov. Rozhodněte, který z následujících případů může nastat a pro možné případy uveďte příklad slov v a w délky alespoň 5.

- a) $HD(v, w) < LD(v, w)$,
- b) $HD(v, w) = LD(v, w)$,
- c) $HD(v, w) > LD(v, w)$.

Př. 8/6: vlastnosti Levenshteinovy vzdálenosti

Označme symbolem $d(x, y)$ Levenshteinovu vzdálenost slov x a y . Víme že, pro tři slova u, v, w platí $d(u, v) = d_1$, $d(v, w) = d_2$. Jakých hodnot může nabývat $d(u, w)$ v závislosti na d_1, d_2 ? Abeceda je pro všechna slova společná.

Př. 8/3: hledání pozměněného slova

V textu nad abecedou $\{a, b, c, d\}$ máme určit všechny výskyty takových podřetězců, které začínají i končí znakem b a zároveň mají od daného vzorku $abbbcdabbbcdab$ Hammingovu vzdálenost větší než 2. Navrhněte konečný nedeterministický automat pro řešení této úlohy.

Př. 8/4: hledání pozměněného slova 2

Konečný automat pro hledání v textu, který hledá všechny podřetězce mající od daného vzorku Levenshteinovu vzdálenost menší než dané k , obsahuje ϵ -přechody. Nakreslete příklad tohoto automatu pro délku vzorku 6 a hodnotu $k = 3$. Dále nakreslete, jak bude tento automat vypadat po odstranění všech ϵ -přechodů.

Př. 8/12: předpona a přípona

Nad abecedou A jsou dány dvě konečné množiny řetězců M_1 a M_2 . Popište, jak sestavíte konečný automat, který přijímá všechna taková slova w nad abecedou A , pro která platí, že alespoň jeden prefix slova w leží v množině M_1 a alespoň jeden suffix w leží v množině M_2 . Připomeňme, že celé slovo se považuje za svůj vlastní prefix i suffix. Sestavte příklad pro $|M_1| = |M_2| = 2$.

Př. 9/1: Hammingovsky blízka slova - dynamicky

Najděte v textu T všechny výskyty řetězců, které mají od vzorku P Hammingovu vzdálenost rovnou nejvýše k . Použijte metodu dynamického programování.

a) $T = ccacbaabccaccbcabccc$, $P = abcba$, $k = 2$,

b) $T = 000111011000101010111110$, $P = 110010$, $k = 3$.

Př. 9/2: Levenshteinovsky blízka slova - dynamicky

Najděte v textu T všechny výskyty řetězců, které mají od vzorku P Levenshteinovu vzdálenost rovnou nejvýše k . Použijte metodu dynamického programování.

a) $T = aacacacbaabbbcbbcacc$, $P = cbbba$, $k = 3$,

b) $T = 010011101000010101011100$, $P = 11100$, $k = 1$.

Př. 9/3: nedeterministické hledání slova z množiny

Sestrojte nedeterministický automat, který v textu nad abecedou A vyhledá právě každé slovo množiny M .

a) $A = \{a, b, c\}$, $M = \{a, b, ba, bc, aaa, bab, ccc, abbc, abcc\}$,

b) $A = \{0, 1\}$,

$M = \{10, 11, 101, 111, 1011, 1101, 10001, 10011, 10111, 11101, 11111\}$.

Př. 9/4: deterministické hledání slova z množiny

Sestrojte deterministický automat, který v textu nad abecedou A vyhledá právě každé slovo množiny M .

a) $A = \{a, b, c\}$, $M = \{a, b, ba, bc, aaa, bab, ccc, abbc, abcc\}$,

b) $A = \{0, 1\}$,

$M = \{10, 11, 101, 111, 1011, 1101, 10001, 10011, 10111, 11101, 11111\}$.