
PAL: 11. cvičení

8. 12. 2022

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

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

Náhodná čísla. Prvočísla.
Modulární umocňování.

Př. 10/2: náhodná čísla

Máte jednu hrací kostku. Popište, jak využijete házení kostkou tak, abyste měli generátor náhodných celých čísel v rozmezí $0 \dots 10$. Všechna čísla $0, 1, 2, \dots, 10$ musí být generována se stejnou pravděpodobností.

Př. 10/3: náhodné uspořádání

Vysvětlete, jak pomocí generátoru náhodných čísel zamícháte do náhodného pořadí seřazené pole čísel. Akce musí proběhnout v čase úměrném délce pole.

Př. 10/4.0: lin. kongruenční generátor náh. čísel

Navrhněte lineární kongruenční generátor tvaru

$x_{n+1} = (ax_n + c) \bmod 10$ tak, aby měl maximální periodu.

Př. 10/4*: lin. kongruenční generátor náh. čísel

Ověřte, zda lineární kongruenční generátor s danými parametry má maximální možnou délku periody.

a) $x_{n+1} = (91x_n + 49) \pmod{600}$,

b) $x_{n+1} = (8x_n + 80) \pmod{49}$,

c) $x_{n+1} = (37x_n + 55) \pmod{144}$,

d) $x_{n+1} = (99x_n + 81) \pmod{113}$.

Př. 10/5: perioda Lehmerova generátoru náh. čísel

Určete délku periody v Lehmerově generátoru, který je dán předpisem $x_{n+1} = ((M - 1) \cdot x_n) \bmod M$, kde M je prvočíslo.

Př. 10/11: největší společný dělitel

Vypočtete největší společný dělitel

a) $GCD(220, 284)$,

b) $GCD\left(\binom{30}{10}, \binom{31}{9}\right)$,

c) $GCD(2^{100}, 100!)$.

Př. 10/12: modulární umocňování

Vypočtete $18^{89} \pmod{11}$.

Skip list. B-stromy.

Př. 11/1: skip list - konstrukce

Sestavte skip list, který je nejprve prázdný a dále do něj vkládáte dané klíče v uvedeném pořadí. Číslo za klíčem v závorce uvádí úroveň (level) klíče, tj. kolikrát byla hozena mince, než padl rub (včetně rubu): 16(3), 23(2), 18(2), 5(2), 15(1), 19(1), 33(1), 11(2), 21(2), 4(1), 22(2), 6(2), 17(4), 10(1), 9(1), 28(4).

Př. 11/2*: skip list - sloučení

Mějme dvě datové struktury skip list délky N . Máme navrhnout efektivní algoritmus, který tyto dva seznamy spojí do seznamu jediného o délce $2N$. Jaká bude jeho asymptotická složitost?

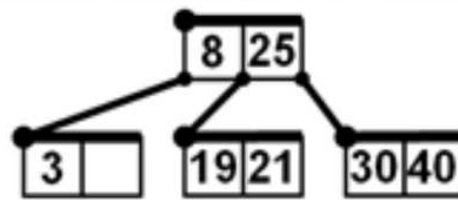
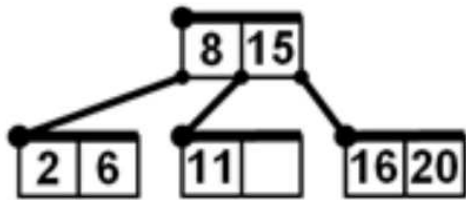
Př. 11/6*: skip list - jiná úroveň

Při implementaci struktury skip list se stalo, že úroveň každého uzlu v operaci Insert je určena jako náhodné číslo z intervalu $[1, \lceil \log_2 N \rceil]$, přičemž se jedná o rovnoměrné rozložení a N je vždy aktuální počet prvků v seznamu.

Odhadněte (pro velké N), nakolik se zhorší asymptotická složitost operací Find/Insert/Delete v této struktuře.

Př. 11/11: B stromy

Do B-stromu znázorněného na levém resp. pravém obrázku vložíme postupně klíče 14, 10, resp. 7, 5. Jaké klíče pak bude obsahovat kořen stromu?



Př. 11/14: konstrukce a destrukce B stromu

B-strom je řádu k , pokud každý jeho uzel, kromě kořene, musí obsahovat alespoň k klíčů a zároveň může obsahovat nejvýše $2k$ klíčů. Vybudujte B-strom řádu 1 tak, že do prázdného stromu vložíte v uvedeném pořadí klíče 25, 13, 37, 32, 40, 20, 22. Dále tento strom zrušte, a to tak, že jednotlivé klíče klíče odstraní v pořadí 13, 25, 40, 22, 20, 37, 32. Nakreslete strom po každé operaci Insert a Delete.

Př. 11/16: plněné B stromy

Je dán a) B-strom, b) B+ strom. Strom je řádu 10 a máme do něj umístit 100 000 klíčů. Jaký je maximální a minimální možný počet uzlů tohoto stromu? Jaká je maximální a minimální možná hloubka tohoto stromu?