KMI/ALS1 Algoritmy a složitost 1

L1: Intro a BST – průměrný případ

Jan Konecny

20. září 2017

Algoritmy a složitost

ALS1 – problematika vyhledávání (BST, hashování, optimální a vyvážené stromy, Catalanova čísla, varianty B-stromů, R-stromy a jejich varianty, NN search, metrické stromy; Pagerank)

ALS2 – problematika těžkých problémů, zejm. algoritmů pro těžké problémy (přibližná řešení těžkých problémů, složitost optimalizačních problémů, aproximační algoritmy pro vybrané těžké problémy, metody jejich návrhu aproximační třídy, randomizované výpočty, heuristiky)

ALS3 – problematika paralelních výpočtů (PRAM model, složitost paralelní výpočtů, třída NC, vyvážené binární stromy, paralelní součet prefixů, přeskakování ukazatelů, technika rozděl a panuj, technika dělení, řetězení výpočtu, 2-3 stromy, akcelerující kaskády. paralelní třídění a zatřiďování, distribuované algoritmy průchodu grafem, konstrukce minimální kostry, volba lídra, kompaktní směrování, Byzantská dohoda)

Zápočet a zkouška

Zápočet – domácí úkoly za body, 75 bodů na zápočet.

Zkouška – klasická ústní zkouška s časem na přípravu odpovědí.

Binární vyhledávací stromy (binary search tree, BST), (opáčko)

Definice

Máme lineárně uspořádanou množinu klíčů.
BST je zakořeněný strom, kde

- uzly jsou označeny klíčem,
- každý uzel má nejvýše dva potomky (nejvýše jednoho levého a nejvýše jednoho pravého).
- pokud má uzel s klíčem k levého potomka s klíčem l, platí $l < k$.
- pokud má uzel s klíčem k pravého potomka s klíčem p, platí $k < p$.

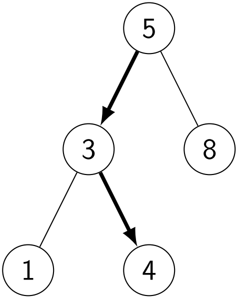
Vyhledávání v BST (opáčko)

Algoritmus Search

Vstup: k – hledaný klíč, R – kořen BST
Výstup: uzel s klíčem k nebo $\emptyset$

If $R = \mathrm{null}$ return $\emptyset$
If $\mathrm{key}(R) = k$ return k
If $\mathrm{key}(R) > k$ return $\mathrm{Search}(k, \mathrm{left}(R))$
If $\mathrm{key}(R) < k$ return $\mathrm{Search}(k, \mathrm{right}(R))$

Search(4, R)



Čas na vyhledávání v nejlepším případě?

Čas na vyhledávání v nejhorším případě?

Čas na vyhledávání v průměrném případě?

Časová složitost vyhledávání

	kořen	průměr	není tam (h)
vyvážený strom	1	?	$h = \log_2(n+1)$
průměrný strom	1	?	$h =?$
degenerovaný strom	1	$n/2$	$h = n$

Domácí úkol (5b)

Zpracujte případ "vyvážený strom, průměr".

Náhodně postavený strom (randomly built BST; RBBST)

Budeme uvažovat zjednodušený případ:

Definition

Náhodně postavený strom o n uzlech, je BST, t.ž.

- pravděpodobnost, že n klíčů bylo vloženo v každém z $n!$ pořadí je stejná;
- ze stromu se nemazalo.

Domácí úkol (5b)

Experimentálně prozkoumejte vliv mazání – odhadněte výšku stromu v těchto případech

- RBBST o 100 uzlech;
- RBBST o 200 uzlech násleně náhodných 100 smazaných.
- 2 vložit, 1 smazat, 200×

Potřebujeme pár pojmů z pravděpodobnosti (I)

Rozdělení pravděpodobnosti (zjednodušeno) Mějme konečnou množinu elementárních jevů Ω. Zobrazení $p : \Omega \to [0, 1]$ t.ž.

$$\sum_{\omega \in \Omega} p(\omega) = 1.$$

nazýváme pravděpodobnost.

Pravděpodobnostní míra:

$$P(A) = \sum_{\omega \in A} p(\omega).$$

Potřebujeme pár pojmů z pravděpodobnosti (II)

Reálná náhodná proměnná $X : \Omega \to \mathbb{R}$.
Indikátor – reálná náhodná proměnná, která nabírá jen hodnot $\{0, 1\}$.

Očekávaná hodnota náhodné proměnné:

$$\mathrm{E}(X) = \sum_{\omega \in \Omega} p(\omega) \cdot X(\omega).$$

Jaká bude očekávaná výška?

Nadefinujeme si tyto tři náhodné proměnné:

- výška RBBST o n prvcích X_n,
- exponenciální výška $Y_n = 2^{X_n}$
- klíč v kořeni RBBST R_n.

Hodnota R_n je se stejnou pravděpodobností kterýkoli z prvků $\{1, \ldots, n\}$.

Pokud $R_n = i$, pak

- levý podstrom je RBBST o $i - 1$ prvcích,
- pravý podstrom je RBBST o $n - i$ prvcích,
- $Y_n = 2 \cdot \max(Y_{i-1}, Y_{n-i})$

Jako krajní případy Y_n máme $Y_1 = 1, Y_0 = 0$.
Dále definujeme náhodné proměnné $Z_{n,1}, Z_{n,2}, \ldots, Z_{n,n}$, kde $Z_{n,i} = I\{R_n = i\}$.

Máme $\Pr\{R_n = i\} = \frac{1}{n}$ pro $i = 1, 2, \ldots, n$, a tedy

$$E[Z_{n,i}] = \frac{1}{n} \text{ pro } i = 1, 2, \ldots, n.$$

Protože právě jedna hodnota $Z_{n,i} = 1$ a všechny ostatní jsou 0, máme taky

$$Y_n = \sum_{i=1}^{n} Z_{n,i}(2 \cdot \max(Y_{i-1}, Y_{n-i})).$$

Ukážeme, že $E[Y_n]$ je polynomická v n,
z toho pak vyplyne, že $E[X_n] = \mathcal{O}(\log n)$.

Náhodná proměnná $Z_{n,i}$ je nezávislá na Y_{i-1} a Y_{n-i}.
Když vybereme $R_n = i$, levý podstrom (s exp. výškou Y_{i-1}) je náhodně postaven z $i-1$ klíčů, které jsou menší než i. Tento podstrom je jako jakýkoli jiný podstrom postavený z $i-1$ prvků:

- pouze počet prvků v něm je závislý na volbě R_n,
- jeho struktura není nijak závislá na volbě R_n.
  Stejně tak pro pravý podstrom.

$$
\begin{aligned}
\mathrm{E}[Y_n] &= \mathrm{E}\left[\sum_{i=1}^{n} Z_{n,i}(2 \cdot \max(Y_{i-1}, Y_{n-i}))\right] \\
&= \sum_{i=1}^{n} \mathrm{E}\left[Z_{n,i}(2 \cdot \max(Y_{i-1}, Y_{n-i}))\right] \\
&= \sum_{i=1}^{n} \mathrm{E}\left[Z_{n,i}\right] \cdot E\left[(2 \cdot \max(Y_{i-1}, Y_{n-i}))\right] \\
&= \sum_{i=1}^{n} \frac{1}{n} \cdot \mathrm{E}\left[(2 \cdot \max(Y_{i-1}, Y_{n-i}))\right] \\
&= \frac{2}{n} \sum_{i=1}^{n} \mathrm{E}\left[\max(Y_{i-1}, Y_{n-i})\right] \leq \frac{2}{n} \sum_{i=1}^{n} \mathrm{E}\left[Y_{i-1}\right] + \mathrm{E}\left[Y_{n-i}\right]
\end{aligned}
$$

$$\frac{2}{n}\sum_{i=1}^{n} \mathrm{E}\left[Y_{i-1}\right] + \mathrm{E}\left[Y_{n-i}\right]$$

V tom výrazu se každý term $\mathrm{E}[Y_0], \mathrm{E}[Y_1], \ldots, \mathrm{E}[Y_{n-1}]$ vyskytuje dvakrát – můžeme ho zjednodušit na

$$E[Y_n] \leq \frac{4}{n}\sum_{i=0}^{n-1} \mathrm{E}[Y_i].$$

Ukážeme, že pro všechna kladná n je to ekvivalentní s

$$E[Y_n] \leq \frac{1}{4}\binom{n+3}{3}.$$

Pro $n = 1$ toto platí

$$1 = Y_1 = \mathrm{E}[Y_1] \leq \frac{1}{4}\binom{1+3}{3} = 1$$

Dále

$$\mathrm{E}[Y_n] \leq \frac{4}{n}\sum_{i=0}^{n-1}\mathrm{E}[Y_i] = \frac{4}{n}\sum_{i=0}^{n-1}\frac{1}{4}\binom{i+3}{3} = \frac{1}{n}\sum_{i=0}^{n-1}\binom{i+3}{3}$$

$$= \frac{1}{n}\binom{n+3}{4} \leq \frac{1}{n}\cdot\frac{(n+3)!}{4!(n-1)!} = \frac{1}{4}\cdot\frac{(n+3)!}{3!\cdot n!} = \frac{1}{4}\cdot\binom{n+3}{3}$$

Platí, že $2^{\mathrm{E}[X_n]} \leq \mathrm{E}[2^{X_n}] = \mathrm{E}[Y_n]$.
Z toho dostáváme, že

$$2^{\mathrm{E}[X_n]} \leq \frac{1}{4}\binom{n+3}{3} = \frac{(n+3)(n+2)(n+1)}{24}.$$

A z toho dostáváme očekávanou výšku

Theorem

Očekávaná výška RBBST o n uzlech je $\mathcal{O}(log(n))$.

Průměrný čas hledání v RBBST

Theorem

Očekávaný počet porovnání při vyhledávání v RBBST o n klíčích je asi

$$2 \ln n \approx 1.386 \log_2 n.$$

Pro jednoduchost budeme uvažovat 'troj-cestné' porovnávání.

Potřebujeme zavést pár pojmů a značení (I)

- C_n – průměrný počet porovnání při úspěšném hledání v BST s n klíči.
- C'_n – průměrný počet porovnání při neúspěšném hledání v BST s n klíči.

$$C_n = 1 + \frac{C'_0 + C'_1 + \cdots + C'_{n-1}}{n} \tag{1}$$

Potřebujeme zavést pár pojmů a značení (II)

Rozšířený binární strom – přidáme zvláštní uzly tam, kde měl původní strom prázdný podstrom.

délka vnější cesty (E – external path length)
Součet vzdáleností kořene od všech vnějších uzlů.

délka vnitřní cesty (I – internal path length)
Součet vzdáleností kořene od všech vnitřních uzlů.

Theorem

Pro BST o n vnitřních uzlech platí, že

$$E = I + 2n. \tag{2}$$

Důkaz.

[Indukcí] $\square$

Pokud předpokládáme, že každý klíč je vyhledáván se stejnou pravděpodobností a že každý z $n+1$ intervalů mezi klíči a vně extrémních hodnot klíčů je stejně pravděpodobný, dostáváme:

$$C_n = 1 + \frac{I}{n} \quad \text{and} \quad C'_n = \frac{E}{n+1}.$$

S použitím (2) dostáváme

$$C_n = \left(1 + \frac{1}{n}\right) C'_n - 1. \tag{3}$$

Z (1) a (3) dostáváme

$$(n+1)C'_n = 2n + C'_0 + C'_1 + \cdots + C'_{n-1}.$$

$$(n+1)C'_n = 2n + C'_0 + C'_1 + \cdots + C'_{n-1}. \tag{4}$$

Zbavíme se rekurence – odečteme od (4) rovnici

$$nC'_{n-1} = 2(n-1) + C'_0 + C'_1 + \cdots + C'_{n-2}.$$

Dostaneme

$$(n+1)C'_n - nC'_{n-1} = 2 + C'_{n-1}.$$

Po úpravě

$$C'_n = C'_{n-1} + \frac{2}{n+1}.$$

$$H_n = 1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n} = \sum_{k=1}^{n} \frac{1}{k} = \ln n + \mathcal{O}(1)$$

poslední krok se dá ukázat aproximací integrály.
Když se dá sumace vyjádřit jako $\sum_{k=m}^{n} f(k)$, kde $f(k)$ je monotónně klesající funkce, můžeme ji aproximovat:

$$\int_{m}^{n+1} f(x)dx \leq \sum_{k=m}^{n} f(k) \leq \int_{m-1}^{n} f(x)dx$$

Dolní hranice:

$$\sum_{k=1}^{n} \frac{1}{k} \geq \int_{1}^{n+1} \frac{1}{x} dx = \ln(n+1)$$

Horní hranice:

$$\sum_{k=2}^{n} \frac{1}{k} \leq \int_{1}^{n} \frac{1}{x} dx = \ln n \quad \text{a tedy} \quad \sum_{k=1}^{n} \frac{1}{k} \leq \ln n + 1$$

Harmonická čísla rekurzivně:

$$H_n = \begin{cases} 0 & \text{pro } n = 0, \\ H_{n-1} + \frac{1}{n} & \text{jinak.} \end{cases}$$

Takže $C'_n = 2H_{n+1} - 2$.
Po dosazení do (3) a zjednodušení dostáváme

$$C_n = 2\left(1 + \frac{1}{n}\right) H_n - 3 \approx 2 \ln n.$$