Algoritmizace: 6. cvičení

Matouš Vrba

25. 10. 2021

Příklad 1

Je zadáno neznámé číslo mezi 1 a 10000. Navrhněte algoritmus, kterým uhodnete zadané číslo v nejmenším počtem otázek. Lze pokládat pouze otázky, na které je odpověď ano, nebo ne. Jaký je maximální počet otázek k uhodnutí zadaného čísla?

Příklad 1

Je zadáno neznámé číslo mezi 1 a 10000. Navrhněte algoritmus, kterým uhodnete zadané číslo v nejmenším počtem otázek. Lze pokládat pouze otázky, na které je odpověď ano, nebo ne. Jaký je maximální počet otázek k uhodnutí zadaného čísla?

Řešením je binární vyhledávání - půlení prohledávaného intervalu.

Příklad 1

Je zadáno neznámé číslo mezi 1 a 10000. Navrhněte algoritmus, kterým uhodnete zadané číslo v nejmenším počtem otázek. Lze pokládat pouze otázky, na které je odpověď ano, nebo ne. Jaký je maximální počet otázek k uhodnutí zadaného čísla?

Řešením je binární vyhledávání - půlení prohledávaného intervalu.

K řešení je potřeba maximálně $\lceil \log_2(1000) + 1 \rceil$ otázek.

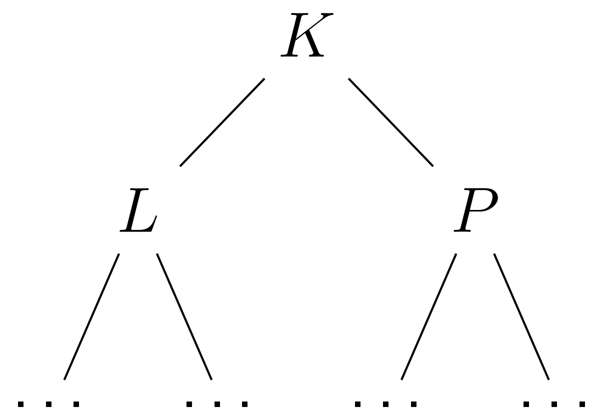
Binární vyhledávací strom

Popište strukturu binárního vyhledávacího stromu (BVS).

Binární vyhledávací strom

Popište strukturu binárního vyhledávacího stromu (BVS).

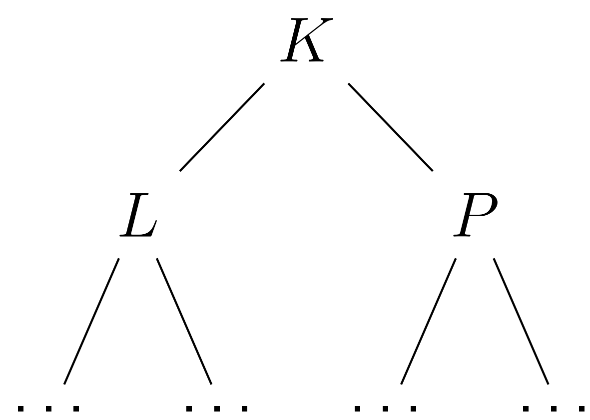
Pro každý podstrom platí $L < K < P$.



Binární vyhledávací strom

Popište strukturu binárního vyhledávacího stromu (BVS).

Pro každý podstrom platí $L < K < P$.



BVS nemusí být *vyvážený* ani *pravidelný*.

Příklad 2

A. Čísla ze zadané posloupnosti postupně vkládejte do prázdného binárního vyhledávacího stromu (BVS), který nevyvažujte. Jak bude vypadat takto vytvořený BVS?

B. Poté postupně odstraňte první tři prvky. Jak bude vypadat výsledný BVS?

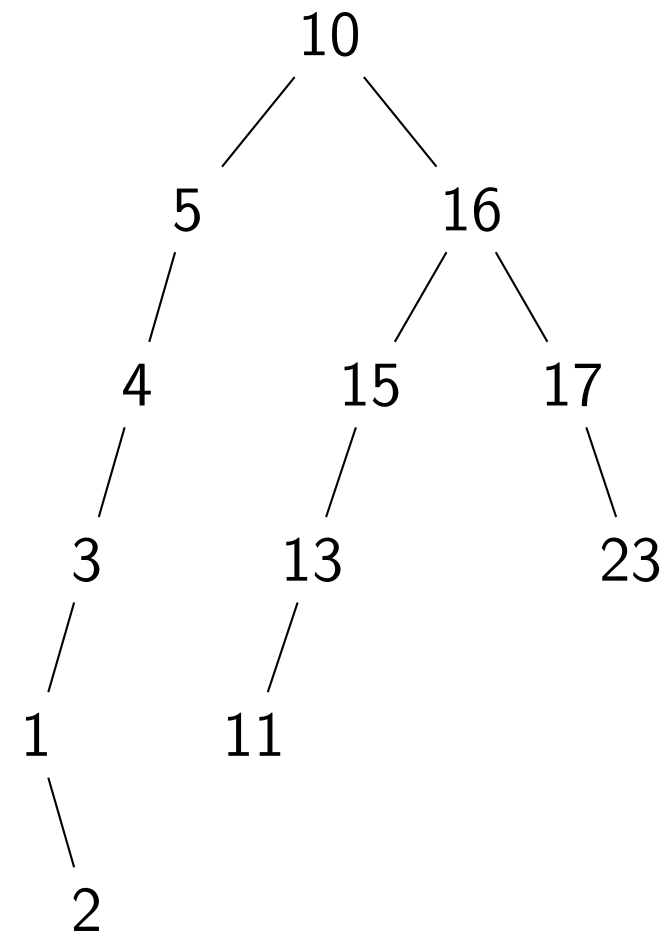
10, 16, 5, 17, 4, 15, 3, 1, 23, 13, 2, 11

Příklad 2

A. Čísla ze zadané posloupnosti postupně vkládejte do prázdného binárního vyhledávacího stromu (BVS), který nevyvažujte. Jak bude vypadat takto vytvořený BVS?

B. Poté postupně odstraňte první tři prvky. Jak bude vypadat výsledný BVS?

10, 16, 5, 17, 4, 15, 3, 1, 23, 13, 2, 11

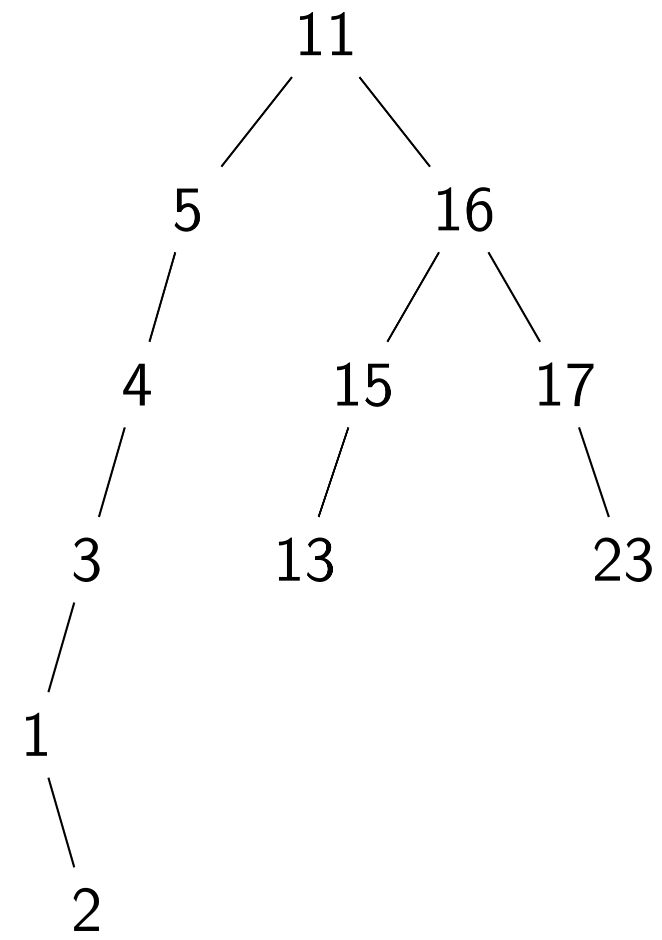


Příklad 2

A. Čísla ze zadané posloupnosti postupně vkládejte do prázdného binárního vyhledávacího stromu (BVS), který nevyvažujte. Jak bude vypadat takto vytvořený BVS?

B. Poté postupně odstraňte první tři prvky. Jak bude vypadat výsledný BVS?

10, 16, 5, 17, 4, 15, 3, 1, 23, 13, 2, 11

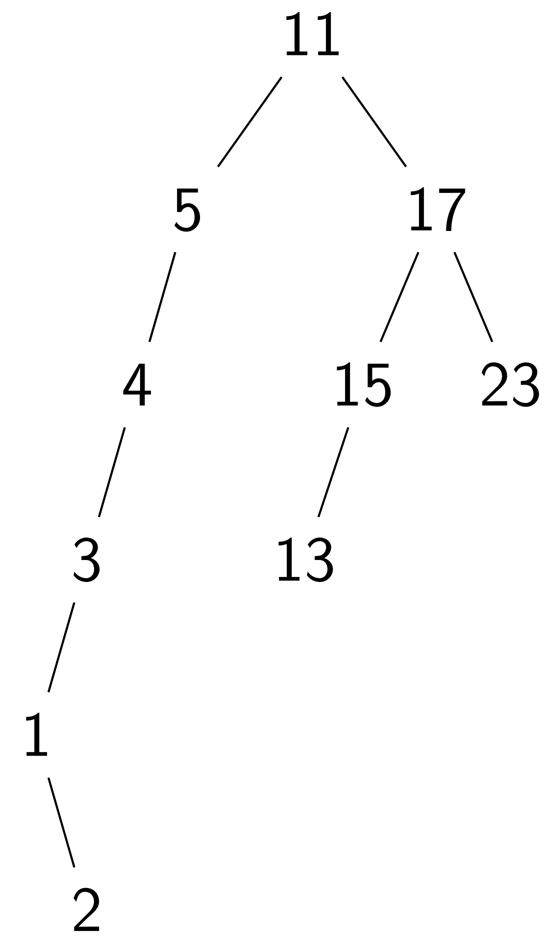


Příklad 2

A. Čísla ze zadané posloupnosti postupně vkládejte do prázdného binárního vyhledávacího stromu (BVS), který nevyvažujte. Jak bude vypadat takto vytvořený BVS?

B. Poté postupně odstraňte první tři prvky. Jak bude vypadat výsledný BVS?

10, 16, 5, 17, 4, 15, 3, 1, 23, 13, 2, 11

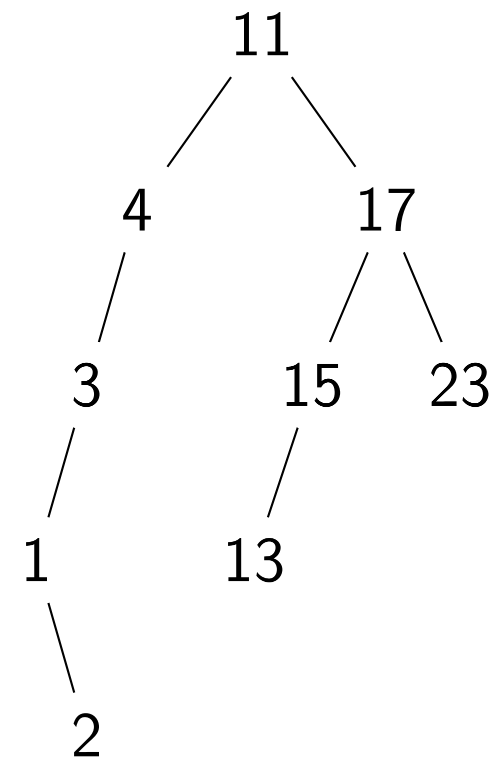


Příklad 2

A. Čísla ze zadané posloupnosti postupně vkládejte do prázdného binárního vyhledávacího stromu (BVS), který nevyvažujte. Jak bude vypadat takto vytvořený BVS?

B. Poté postupně odstraňte první tři prvky. Jak bude vypadat výsledný BVS?

10, 16, 5, 17, 4, 15, 3, 1, 23, 13, 2, 11



Příklad 3

V jakém pořadí vypíšeme prvky binárního vyhledávácího stromu, pokud ho projdeme inorder?

Příklad 3

V jakém pořadí vypíšeme prvky binárního vyhledávácího stromu, pokud ho projdeme inorder?

V uspořádaném pořadí (od nejmenšího k největšímu).

Příklad 4

Složitost operace Insert v obecném BVS s n uzly a hloubkou d je vždy

A. $\mathrm{O}\left(\log(n)\right)$
B. $\Theta\left(\log(n)\right)$
C. $\mathrm{O}\left(n\right)$
D. $\Theta\left(n\right)$
E. $\mathrm{O}\left(d\right)$
F. $\Theta\left(d\right)$
G. $\mathrm{O}\left(\log(d)\right)$
H. $\Theta\left(\log(d)\right)$

Příklad 4

Složitost operace Insert v obecném BVS s n uzly a hloubkou d je vždy

A. $\mathrm{O}(\log(n))$
B. $\Theta(\log(n))$
✓ $\mathrm{O}(n)$
D. $\Theta(n)$
E. $\mathrm{O}(d)$
F. $\Theta(d)$
G. $\mathrm{O}(\log(d))$
H. $\Theta(\log(d))$

Poznámka: Platí také pro operace Find a Delete.
Pro *vyvážený* BVS je složitost těchto operací $\mathrm{O}(\log(n))$

Příklad 5

Předpokládejme, že binární vyhledávací strom obsahuje přirozená čísla mezi 1 a 1000. Která z následujícíh sekvencí navštívených uzlů nemůže nastat, pokud hledáne klíč 363?

A. 2, 252, 401, 398, 330, 363
B. 399, 387, 219, 266, 382, 381, 278, 363
C. 3, 923, 220, 911, 244, 898, 258, 362, 363
D. 4, 924, 278, 347, 621, 299, 392, 358, 363
E. 5, 925, 202, 910, 245, 363

Příklad 5

Předpokládejme, že binární vyhledávací strom obsahuje přirozená čísla mezi 1 a 1000. Která z následujícíh sekvencí navštívených uzlů nemůže nastat, pokud hledáne klíč 363?

A. 2, 252, 401, 398, 330, 363
B. 399, 387, 219, 266, 382, 381, 278, 363
C. 3, 923, 220, 911, 244, 898, 258, 362, 363
✓ 4, 924, 278, 347, 621, 299, 392, 358, 363
E. 5, 925, 202, 910, 245, 363

Příklad 6

Mějme binární vyhledávací strom s n uzly. Jaká je asymptotická složitost operace, která spočítá klíče, jejichž hodnota je menší, než x?

Příklad 6

Mějme binární vyhledávací strom s n uzly. Jaká je asymptotická složitost operace, která spočítá klíče, jejichž hodnota je menší, než x?

Lze použít procházení inorder pro nalezení prvního prvku, jehož hodnota je větší, než x. Jednoduše spočítáme počet prohledaných uzlů. Složitost tohoto přístupu je $\mathrm{O}(n)$.

Interpolační hledání

Jaké jsou předpoklady pro použití interpolačního hledání?
Popište metodu interpolačního hledání.
Jaká je asymptotická složitost interpolačního hledání?

Interpolační hledání

Jaké jsou předpoklady pro použití interpolačního hledání?
Popište metodu interpolačního hledání.
Jaká je asymptotická složitost interpolačního hledání?

Prohledávané pole musí být seřazené.
Prvky pole musí být číselné (narozdíl od binárního prohledávání, kterému stačí, aby byly prvky porovnatelné).
Interpolační metoda funguje tím lépe, čím blíže rovnoměrnému rozložení jsou hodnoty v poli.

Interpolační hledání

Jaké jsou předpoklady pro použití interpolačního hledání?
Popište metodu interpolačního hledání.
Jaká je asymptotická složitost interpolačního hledání?

Pole:

$$\left(3,\quad 5,\quad 8,\quad 10,\quad 14,\quad 15,\quad 16,\quad 22,\quad 25,\quad 30\right),$$

hledaný prvek: 16.

Interpolační hledání

Jaké jsou předpoklady pro použití interpolačního hledání?
Popište metodu interpolačního hledání.
Jaká je asymptotická složitost interpolačního hledání?

1. Pokud je prohledávaná část pole prázdná (neobsahuje žádné prvky), nenalezli jsme hledaný prvek a skončíme.
2. Lineární interpolací mezi prvním a posledním prvkem prohledávané části pole odhadneme pozici hledaného prvku.
3. Pokud je na odhadnuté pozici hledaný prvek, končíme.
4. Pokud je prvek na odhadnuté pozici menší, než hledaný prvek, nastavíme poslední prvek prohledávané časti na odhadnutou pozici a opakujeme od kroku 1.
5. Pokud je prvek na odhadnuté pozici větší, než hledaný prvek, nastavíme první prvek prohledávané časti na odhadnutou pozici a opakujeme od kroku 1.

Interpolační hledání

Jaké jsou předpoklady pro použití interpolačního hledání?
Popište metodu interpolačního hledání.
Jaká je asymptotická složitost interpolačního hledání?

Asymptotická složitost interpolačního prohledávání je obecně $\mathrm{O}(n)$ (takže horší, než binární prohledávání).
Pokud jsou data v poli rozložena rovnoměrně v lineárním měřítku, je asymptotická složitost interpolačního prohledávání $\mathrm{O}(\log(\log(n)))$.

Samostatná práce

Řešte úlohy 4, 5, 6, 9, 14, 15.

Řešení zašlete do konce dne na `matous.vrba@fel.cvut.cz`
s předmětem ALG06.