

4 Spojové struktury

Wednesday, 19 January 2022 09:02

Spojové struktury, seznamy, stromy.

- spojová struktura
 - struktura obsahující soubor uzlů propojených pomocí referencí (odkazů) - relativní umístění vůči ostatním uzlům
 - a. zřetěžená
 - uzel má referenci na další uzel ve struktuře
 - b. nezřetěžená
 - přístup k dalšímu uzlu pomocí dopočítávání adresy v paměti
- spojový seznam
 - spojová struktura dynamické délky
 - vstupní uzel = head (první prvek)
 - prázdný seznam → head = null
 - využití u zásobníků, front, dynamického alokování paměti
 - a. jednosměrný
 - uzel má referenci na další
 - další uzel posledního uzlu = null
 - b. dvousměrný
 - uzel má referenci na další a předchozí
 - předchozí uzel prvního uzlu = null
 - c. cyklický
 - poslední uzel ukazuje na první
 - další uzel posledního uzlu = head
 - složitost
 - přidání
 - na začátek: $O(1)$
 - na konec: $O(1)$ - pokud je uložena reference konce, jinak $O(n)$
 - na index: $O(n)$
 - odebrání
 - headu: $O(1)$
 - libovolného uzlu: $O(n)$
 - indexace: $O(n)$
 - vyhledávání: $O(n)$
- graf
 - uzly propojeny hranami
 - a. kružnice
 - N uzlů, N hran
 - b. cesta
 - N uzlů, $N-1$ hran
 - c. strom
 - cesta, kde může mít uzel více sousedů, bez kružnic
 - d. úplný graf
 - N uzlů, $(N^2-N)/2$ hran (každý s každým)
- strom
 - výchozí uzel = root (nejvyšší prvek)
 - uzly mají několik následníků (v binárním stromu dva)
 - využití v třídících a vyhledávacích algoritmech, routovacích tabulkách, kompresních algoritmech a blockchainu
 - přirozený model rekurze
 - list (*leaf*)
 - uzel bez potomků

- a. halda (*heap*)
 - vyvážený binární strom
 - pro každý uzel platí, že jeho rodič nese stejnou nebo vyšší hodnotu
- b. BST (*binary search tree* - binární vyhledávací strom)
 - efektivní práce s daty
 - pravá větev vždy obsahuje prvky větší než hodnota uzlu
 - s vyvažováním: AVL strom (hloubka levého a pravého stromu se liší max. o 1)
- c. B-strom
 - v uzlech lze uložit více hodnot
- vlastnosti:
 - N-arity
 - kolik může mít uzel maximálně potomků (binární - max. 2)
 - hloubka
 - délka cesty od kořene k uzlu
 - výška (hloubka stromu)
 - maximální hloubka
 - šířka
 - počet uzlů na stejné úrovni (se stejnou hloubkou)
 - pravidelnost
 - každý uzel má 0 nebo N následníků
 - vyváženost
 - hloubka stromu se liší max. o 1
 - úplná pravidelnost
 - strom, který má všechny hloubky plné
tj. všechny listy jsou v hloubce h a je jich přesně 2^h
- složitost u BST
 - přidání, odebrání, vyhledávání: $O(\log n)$ až $O(n)$
- akce:
 - mazání
 - uzel bez potomků (list)
 - ◆ smazání odkazu u předka uzlu
 - uzel s jednou větví (X, potomek L/R)
 - ◆ posunutí větve tak, že rodič prvku X nastaví hodnotu left/right na referenci na potomka L/R
 - uzel se dvěma větvemi (X, potomci L a R):
 - ◆ posunutí levé větve
 - ◆ rekurzivně se jede doprava, dokud se nenajde uzel L, který nemá pravého potomka
 - ◆ pravým potomkem tohoto uzlu L se nastaví pravý následník P mazaného uzlu X
 - procházení
 - do hloubky
 - do šířky
 - pre-order (uzel, levý podstrom, pravý podstrom)
 - in-order (levý podstrom, uzel, pravý podstrom)
 - post-order (levý podstrom, pravý podstrom, uzel)