

Binarne drzewo przeszukiwań z łączem do rodzica + algorytm równoważący DSW

Spis treści

| | |
|--------------------------------|---|
| 1. Używanie biblioteki..... | 1 |
| Klasa Node..... | 1 |
| Klasa Btree..... | 1 |
| 2. Jak działa biblioteka | 1 |
| Klasa Node..... | 1 |
| Klasa Btree..... | 1 |
| 3. Bibliografia..... | 2 |

1. Używanie biblioteki

Klasa Node

- Konstruktor inicjowany wartością przechowywaną

Klasa Btree

Klasa przechowuje wskaźnik do korzenia drzewa w zmiennej top.

- Konstruktor opcjonalnie inicjowany elementem klasy Node
- print_indented – funkcja wypisująca elementy drzewa na ekranie
- insert – wstawia element klasy Node i zwraca nowy korzeń
- search – zwraca węzeł lub None zawierający element podany jako argument
- delete – usuwa z drzewa element klasy Node podany jako argument
- DSW – równoważy drzewo

2. Jak działa biblioteka

Klasa Node

- data – dane
- left – wskaźnik do lewego dziecka
- right – wskaźnik do prawego dziecka
- parent – wskaźnik do rodzica

Klasa Btree

- print_indented – funkcja wypisująca elementy drzewa na ekranie

Funkcja rekurencyjnie wypisuje elementy w kolejności preorder.

- insert – wstawia element i zwraca nowy korzeń

Funkcja rekurencyjnie wyszukuje pierwsze wolne miejsce aby wstawić nowy element zgodnie z zasadą po prawej stronie drzewa większe elementy. Jeśli wstawiany element jest duplikatem funkcja go on ignorowany.

- search – zwraca węzeł lub None zawierający element podany jako argument

Funkcja rekurencyjnie poszukuje węzła który przechowuje dane podane jako argument.

- delete – usuwa z drzewa element klasy Node podany jako argument

Podczas usuwania funkcja rozważa trzy przypadki:

- Węzeł nie ma dzieci. Węzeł możemy bez problemu usunąć.
- Węzeł ma jedno dziecko. Wtedy podnosimy dziecko do poziomu usuwanego węzła.
- Węzeł ma dwoje dzieci. Tutaj musimy wykonać operację podnoszenia następników odpowiedniego syna.
 - `_btree_transplant`

Funkcja wstawia jedno poddrzewo w miejsce drugiego.

- `_btree_find_min`

Funkcja zwraca element klasy Node przechowujący najmniejszą wartość. W tym celu porusza się po drzewie aż do liścia w lewą stronę.

- DSW – równoważy drzewo

Funkcja pierw uruchamia metodę `createSpine()` a następnie `CreateWeightedTree`.

- `_spine_height` – zwraca wysokość „Kręgosłupa”

Funkcja zakłada że podane drzewo ma w swojej strukturze tylko prawych dzieci. Liczy wszystkie węzły i zwraca wynik.

- `_left_rotation` – rotacja drzewa w prawo

Funkcja wykonuje na drzewie rotację w lewo przepinając odpowiednio wskaźniki do rodzica oraz prawego i lewego dziecka.

- `_createSpine` – tworzy „Kręgosłup” czyli zmienia drzewo w listę

Funkcja z drzewa tworzy drzewo w którym węzły mają tylko prawych potomków.

- `_createWeightedTree` – tworzy idealnie wyważone drzewo z listy

Funkcja wykorzystując obroty w lewo z drzewa w formie „kręgosłupa” tworzy idealnie wyważone drzewo.

3. Bibliografia

- Drzewo binarne: <http://users.uj.edu.pl/~ufkapano/algorytmy/lekcja09/btree1.html>
- Algorytm DSW: https://pl.wikipedia.org/wiki/Algorytm_DSW