

# 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
- btree\_print\_indented – funkcja rysująca drzewo na ekranie
- btree\_insert – wstawi element klasy Node i zwraca nowy korzeń
- btree\_search – zwraca węzeł lub None zawierający element podany jako argument
- btree\_delete – usuwa z drzewa element klasy Node podany jako argument
- btree\_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

- btree\_print\_indented – funkcja rysująca drzewo na ekranie
- btree\_test\_print\_indented – funkcja wypisująca elementy w specjalnej kolejności do testów
- bst\_insert – wstawi 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.

- `btree_search` – zwraca węzeł lub `None` zawierający element podany jako argument

Funkcja rekurencyjnie poszukuje węzła który przechowuje dane podane jako argument.

- `btree_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.

- `right_rotation` – rotacja drzewa w prawo

Funkcja wykonuje na drzewie rotację w prawo 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 prawo 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](https://pl.wikipedia.org/wiki/Algorytm_DSW)