

- ! Drzewa:
- ! BST " preorder, inorder, postorder, nastepnik, poprzednik, usuwanie, wstawianie, rotacje
- ! Algorytm DSW
- # RBT "
 - # czarna wysokosc, [rbt.md](#)
 - # zasady - czarna wysokosc nie tyczy sie tylko korzenia tutaj,
 - ! uzasadnienie logarytmicznosci,
 - ~~usuwanie(nie bylo na prezce i ponoc nie pokazywal mowil za ciezkie),~~
 - # dodawanie,
 - # rotacje/fixupy
 - ! pseudokod dodawania z fixupami/rotacjami [rbt.cpp Source Code](#)
- ! B- drzewo " wstawianie, usuwanie
- ! B+ drzewo " wstawianie, usuwanie
- ! Prefiskowe b+ drzewo (to jako ciekawostka jakas pewnie)
- ! B* drzewo - 2 slajdy, wstawianie
- ! (a,b)-drzewo " 2 slajdy, wstawianie
- ! ~~Trie (drzewo prefiksowe) - 1 slajd tylko~~
- ! k-d drzewo - na prezce 1D " wywazanie, wstawianie, usuwanie
- ! ~~R-drzewo " (jako ciekawostka)~~
- ! Splay
- ! Threaded trees (przed bst, pseudokod da\$ pewnie jako wprowadzenie do bst)
- ! ~~AVL~~ mowil ze nie bedzie
- ! Kopce: kinda
- ! wstawianie, usuwanie
- ! Binarny (Min/Max)
- # Dwumianowy " \$czenie dw-ch kopc-w dwumianowych, zmniejszenie wartosci dowolnego wezla
- # Fibonacciego " laczenie, zmniejszanie, usuwanie
- # sprz&' ony
- # Min-Max
- ! Drzewo dwumianowe (jest w prezce o kopcach) " wlasnosci
- ! Haszowanie:
- # cuckoo
- ! Robin Hooda " niby na prezce nie bylo ale gdzies tam bylo omawiane i sie tu uczylismy po cos
- ! Coalescened

- # Podwójne
- # Liniowe
- ! Kwadratowe ? - nie było na prezce
- ! Kubełkowe (Metoda śa(uchowa)
- ! Hopscotch ? - nie było na prezce
- ! Sortowania:
- # Quicksort(koncept), pivot
 - # pseudocode
 - # complexity
- # Radix Sort(koncept), od cyfry jedności, z pomocniczym algorytmem
 - ! pseudocode
 - ! complexity
- # Merge Sort(koncept), dzielimy i łączymy
 - ! pseudocode
 - ! complexity
- # Heapsort(koncept)
 - ! pseudocode
 - ! complexity
- # Counting(koncept)
 - ! pseudocode
 - ! complexity
- # Bucket(koncept)
 - ! pseudocode
 - ! complexity
- # Bubble(koncept)porównywać po kolei i zmieniać (z optymalizacją)
 - ! pseudocode
 - ! complexity
- # Insertion(koncept)sprawdzamy po lewej stronie czy liczba rozpatrywana jest większa od liczb po lewej jeśli nie to wstawiamy za najmniejszą liczbą najbliższą jej
 - ! pseudocode
 - ! complexity
- # Selection(koncept)find minimum and swap
 - ! pseudocode
 - ! complexity
- # Shell(koncept) - nie powinno być

- ! pseudocode
- ! complexity
- ! Programowanie dynamiczne, inne:
 - # Odleg ϕ) * Levensteina
 - # Problem wydawania reszty
 - # Algorytm Floyd-Warshalla
 - ! Ci ϕ g Fibonacciego (Obliczanie)
 - ! Algorytm Kruskala
 - # Algorytm Grahama (Graham-Scan)
 - ! FFT/DFT nawet dla 8
- ! Inne:
 - ! BigO notation
 - ! Pseudokod
 - ! Lista wska+nikowa (lista liniowa)
 - ! Skip list (Lista z przeskokami) - usuwanie, wstawianie
 - ! Array list