Sprawozdanie

Algorytmy I Struktury Danych

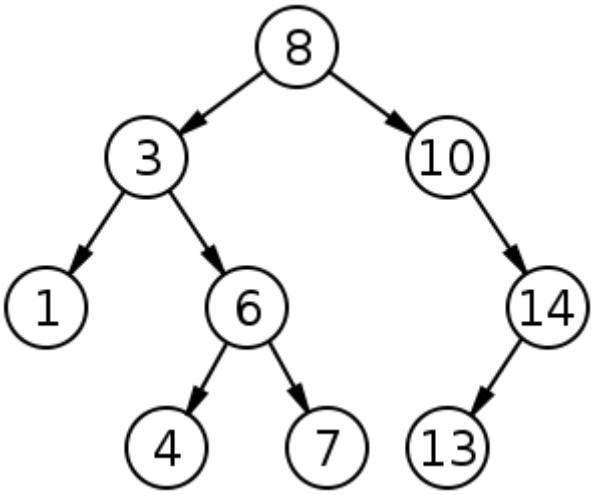
Lista 4

Krzysztof Strzała 236503

1.Wprowadzenie

1. Drzewo BST

https://en.wikipedia.org/wiki/Binary_search_tree



Drzewo wyszukiwań binarnych w którym każdy węzeł cechuje

- jeżeli jest prawym synem to ma większą wartość od ojca.

- jeżeli jest lewym synem to jego wartość jest mniejsza od ojca.

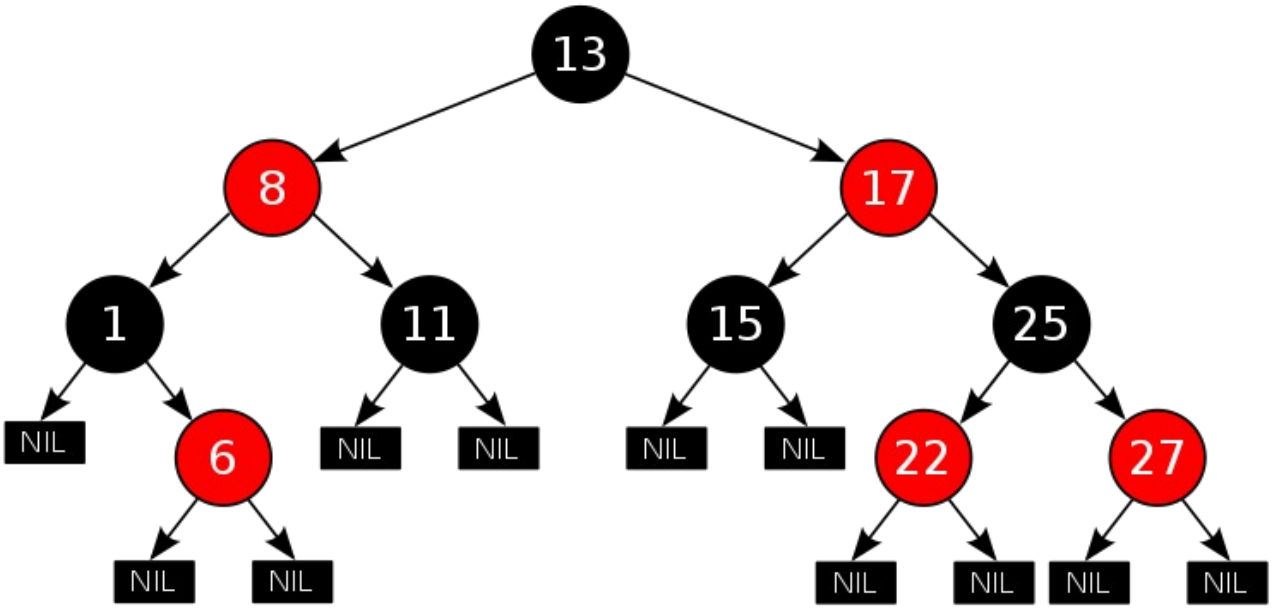
Dla zrównoważonego drzewa złożoność operacji insert , delete jak I search możemy określić rzędu O(log n). W pesymistycznym przypadku złożoność jest rzędu O(n) , gdy węzły drzewa ułożone są w lini.

2. Drzewo RB

https://en.wikipedia.org/wiki/Red%E2%80%93black_tree

Drzewo czerwono-czarne które różni się tylko tym że kolorujemy węzły na kolor czarny I czerwony , a jego warunkiem jest brak sąsiadujących ze sobą 2 czerwonych wierzchołków. Niesie to za sobą zwiększony czas o “naprawy drzewa” w przypadku operacji insert I delete. Porównując ze zrównoważonym drzewem BST drzewo będzie miało większą złożoność czasową , jednakże

porównując to z drzewo z drzewem BST którego wierzchołki układają się w lini , uzyskamy mniejszą złożoność.



3. Drzewo Splay

https://en.wikipedia.org/wiki/Splay_tree

Splay to rozszerzenie drzewa BST operacje splay wykonywaną przy operacji search I delete. Splay polega na umieszczeniu szukanego / usuwanego / dodawanego klucza w korzeniu przy zachowaniu porządku drzewa BST.

Drzewa tego typu przewyższają drzewa RB w przypadku zwiększonej ilości wyszukiwań podobnych kluczy nad insertami.

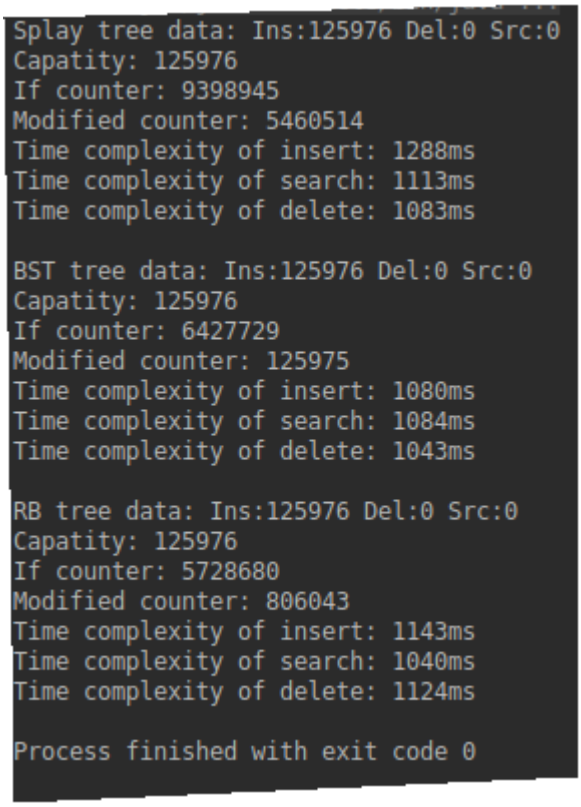
Poniższy przykład obrazuje jak wygląda złożność w przypadku wyszukiwania podobnych kluczy wiele razy.

```
SPLAY DATA WITH 1000000 NUMBERS
insert time =2278
find time =481
removal time = 1254

BST TREE DATA WITH 1000000 NUMBERS
insert time =780
find time =7617
removal time =1002

REDBLACK TREE DATA WITH 1000000 NUMBERS
insert time =939
find time =821
removal time =582
```

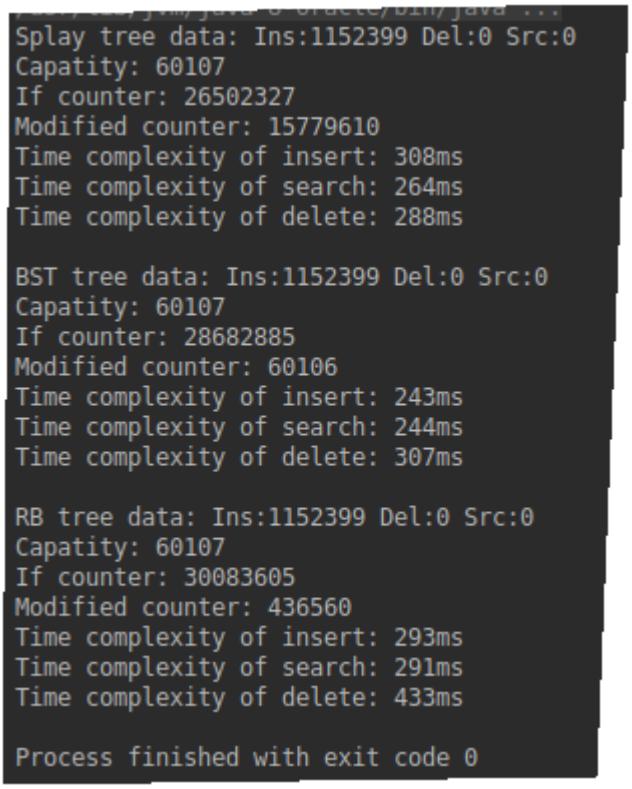
2.Testy

Testy z plików podanych na stronie przeprowadzono w następujący sposób:

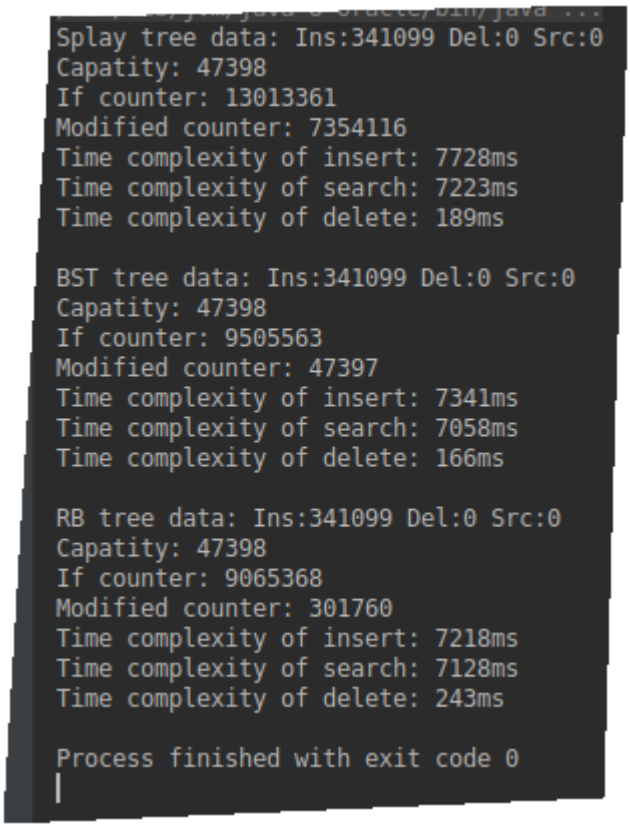
- stworzono tablice złożoną z wszystkich słów w danym liku tekstowym.
- losowe elementy z tablicy zostały dodawane do drzew , tak długo aż tablica była pusta.
- wykonano operacje search w losowej kolejności o każdy element z pliku
- usunięto w losowej kolejności każdy element



Dane dla unikatowych ciągów posortowanych aspell_wordlist.txt



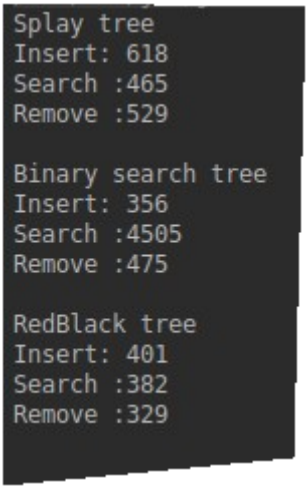
Dane dla pliku KJB.txt

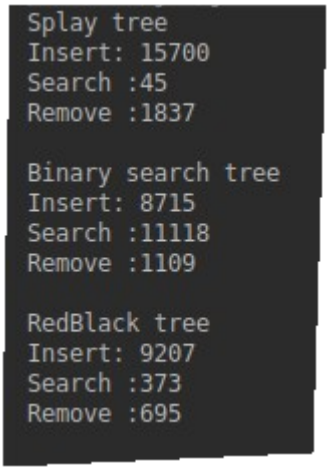


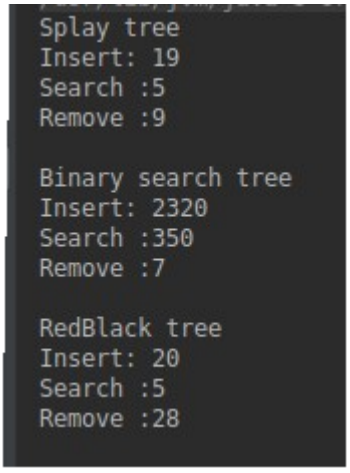
Dane dla książki Anna Karenina (zamiast Łotr).

Kolejne przykłady przeprowadzono na zbiorach liczb :

oznaczenia – I – ilość insertów , S – ilość wyszukiwań kluczy , D – ilość usunięć kluczy







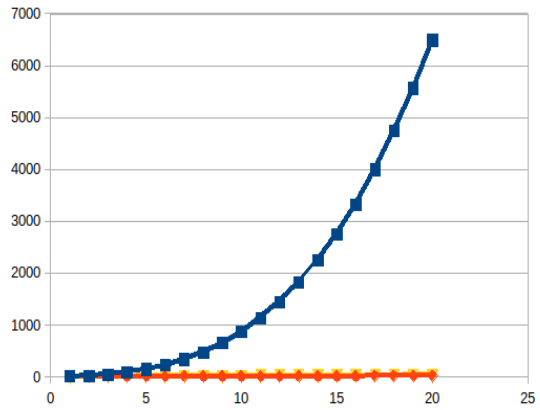
1.Średni czas dla przypadku gdy 10I = 10D = S

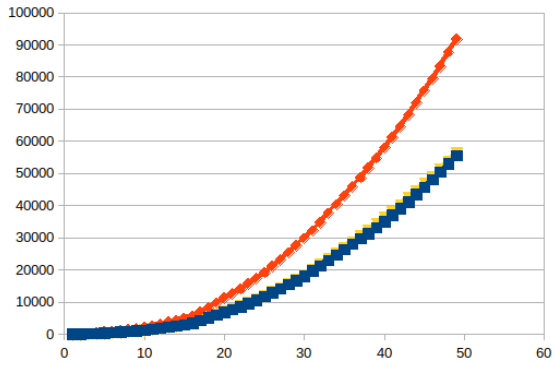
2. Średni czas dla przypadku gdy I = 10D = 10S

3. Średni czas dla przypadku gdy I = R = 10S , lecz inserty są kolejnymi liczbami całkowitymi (1,2,3,4…)

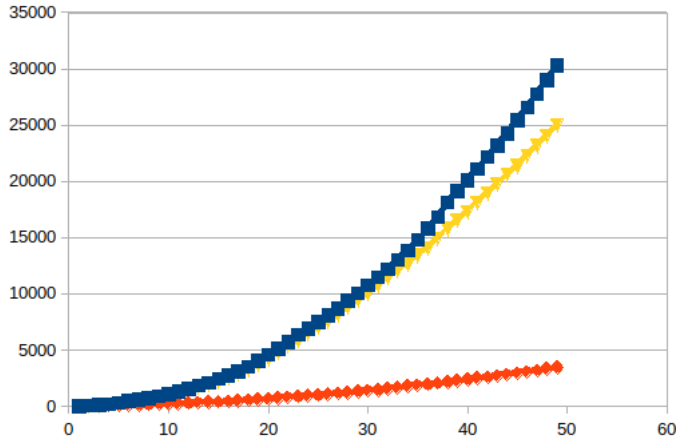
1. Porównanie struktur dla dodawania wartości o liniowych wartościach klucza (1,2...n):

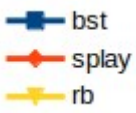
2. Porównanie struktur dla dodawania losowych wartości.





3. Porównanie struktur dla 50 razy częściej występującej operacji search.





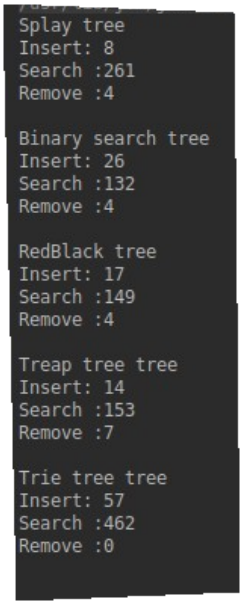
**** Porównanie z innymi strukturami:

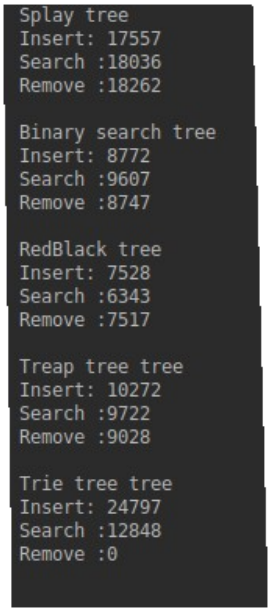
```
Splay tree
Insert: 4
Search :6
Remove :6

Binary search tree
Insert: 120
Search :111
Remove :113

RedBlack tree
Insert: 19
Search :3
Remove :6

Treap tree tree
Insert: 6
Search :4
Remove :3

Trie tree tree
Insert: 24
Search :23
Remove :0
```

```
Splay tree
Insert: 19
Search :4
Remove :6

Binary search tree
Insert: 5
Search :3
Remove :5

RedBlack tree
Insert: 4
Search :7
Remove :27

Treap tree tree
Insert: 25
Search :9
Remove :28

Trie tree tree
Insert: 76
Search :13
Remove :0
```

```
Splay tree
Insert: 52
Search :11
Remove :10

Binary search tree
Insert: 56
Search :16
Remove :11

RedBlack tree
Insert: 48
Search :5
Remove :7

Treap tree tree
Insert: 71
Search :15
Remove :15

Trie tree tree
Insert: 148
Search :17
Remove :0
```

```
Splay tree
Insert: 56
Search :3
Remove :63

Binary search tree
Insert: 37
Search :5
Remove :53

RedBlack tree
Insert: 37
Search :5
Remove :45

Treap tree tree
Insert: 50
Search :5
Remove :48

Trie tree tree
Insert: 115
Search :13
Remove :0
```





1. Dane dodawane liniowo
2. Mała ilość danych
3. 10 razy więcej insertów niż pozostałych operacji.
4. 10 razy więcej insertów I usunięć niż pozostałych operacji.
5. 100 razy więcej insertów I wyszukiwań losowych wartości z tablic niż usunięć.
6. 100 razy więcej insertów I wyszukiwań podobnych wartości z tablic niż usunięć.
7. 100 razy więcej wyszukań niż innych operacji.
8. Duża ilość danych