

# Sprawozdanie z ćwiczenia laboratoryjnego V

## Drzewa BST i AVL

Bartłomiej Ankowski

17.04.2015

# Spis treści

<b>1</b>	<b>Wstęp</b>	<b>1</b>
<b>2</b>	<b>Złożoność obliczeniowa</b>	<b>1</b>
2.1	Drzewo AVL . . . . .	1
2.2	Drzewo BST . . . . .	2
2.2.1	Zapis . . . . .	2
2.2.2	Odczyt . . . . .	2
<b>3</b>	<b>Wyniki pomiarów</b>	<b>3</b>
3.1	Odczyt . . . . .	3
3.2	Zapis . . . . .	5
<b>4</b>	<b>Wnioski</b>	<b>7</b>

## 1 Wstęp

Celem tego laboratorium było zamodelowanie dwóch struktur danych: Drzewo BST i AVL. Następnie został wykonany pomiar czasów zapisu i odczytu danych dla tych struktur.

## 2 Złożoność obliczeniowa

### 2.1 Drzewo AVL

Drzewo AVL posiada mechanizm autobalansujący strukturę drzewa. Dzieje się tak za sprawą rotacji: lewych, prawych bądź mieszanych wykonywanych w

momencie, gdy moduł współczynnika balansujący będzie większy od 1. Współczynnik ten jest wyznaczany jako różnica lewego i prawego poddrzewa. Drzewo jest wyważone jeśli dla każdego węzła liczba węzłów w jego lewym i prawym poddrzewiu różni się maksymalnie o jeden. Złożoność obliczeniowa wynikająca jest nieco większa niż w przypadku drzewa BST, lecz własności drzewa AVL powodują, że czas odczytu i zapisu nie powinien przekroczyć  $1,44\log_2 n$ , gdzie  $n$  jest liczbą węzłów. Zatem złożoność obliczeniowa operacji zapisu i odczytu wynosi  $O(\log_2 n)$ .

## 2.2 Drzewo BST

### 2.2.1 Zapis

Algorytm dodawania węzła w sposób rekurencyjny wykonuje się od korzenia w stronę liści, poruszając się w lewo lub w prawo, w zależności od wyniku porównania z napotkanymi węzłami. W momencie dotarcia do jednego z liści następuje ostateczne porównanie wartości, po czym nowy węzeł staje się lewym bądź prawym synem węzła, który był liściem. Złożoność tej operacji wynosi  $O(h)$ , gdzie  $h$  jest wysokością drzewa, wyrażającą się wzorem  $\log_2 n$ , zatem całkowita złożoność wynosi  $O(\log_2 n)$ .

Konstrukcja takiego drzewa nie posiada żadnych mechanizmów utrzymujących strukturę w zakładanym porządku. Może zatem dojść do sytuacji, że drzewo utraci swoją strukturę i zdegeneruje się do postaci listy. W takim przypadku złożoność obliczeniowa wyniesie  $O(n)$ , ponieważ dodanie nowego węzła wymusi przejście wszystkich elementów od korzenia, aż do liścia.

### 2.2.2 Odczyt

Algorytm Znajdowania interesującego nas węzła jest proporcjonalny do wysokości drzewa. Wykonywane są rekurencyjne wywoływania dla kolejnych lewych bądź prawych węzłów, aż do momentu natrafienia na szukaną wartość. Złożoność tej operacji wynosi zatem  $O(\log_2 n)$ .

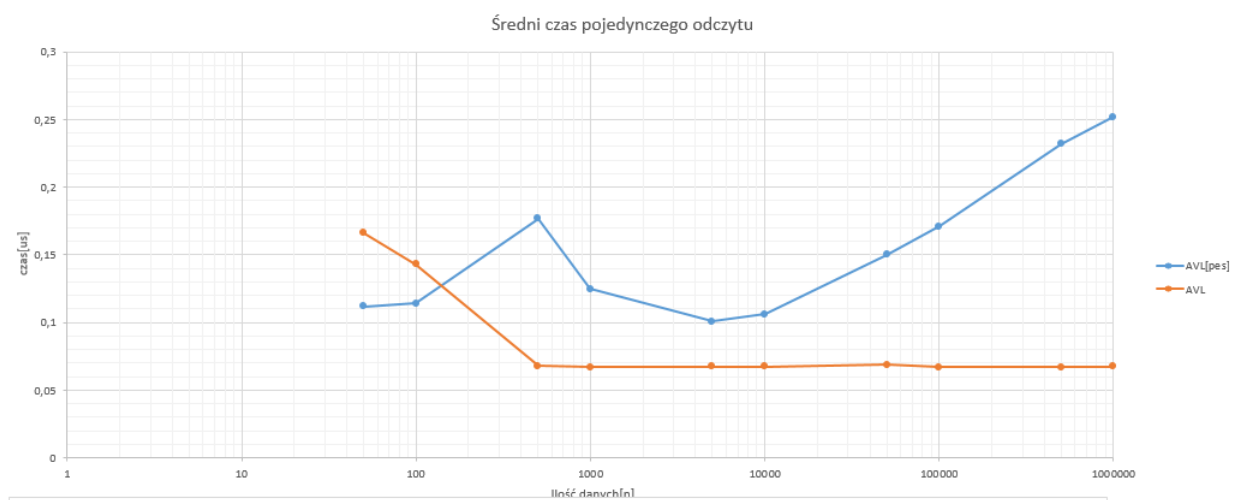
W przypadku, gdy doszło do zdegenerowania struktury podczas zapisu, przełoży się to również na złożoność obliczeniową odczytu. Złożoność w takim przypadku wyniesie również  $O(\log_2 n)$ .

## 3 Wyniki pomiarów

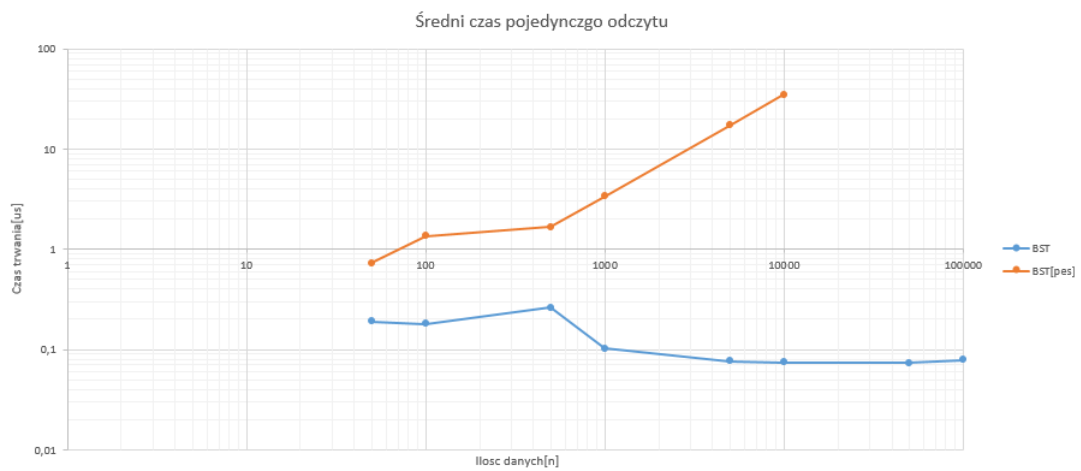
### 3.1 Odczyt

Ilość danych[n]	AVL pes[us]	AVL[us]	BST[us]	BST pes[us]
50	0,112	0,166	0,190	0,724
100	0,114	0,143	0,180	1,353
500	0,177	0,068	0,261	1,660
1000	0,125	0,067	0,102	3,352
5000	0,101	0,0673	0,0768	17,21
10000	0,106	0,0677	0,0743	34,93
50000	0,150	0,0689	0,07338	-
100000	0,171	0,0677	0,0783	-
500000	0,232	0,0672	-	-
1000000	0,252	0,0677	-	-

Tablica 1: Wyniki odczytu pojedynczego elementu dla drzew AVL i BST



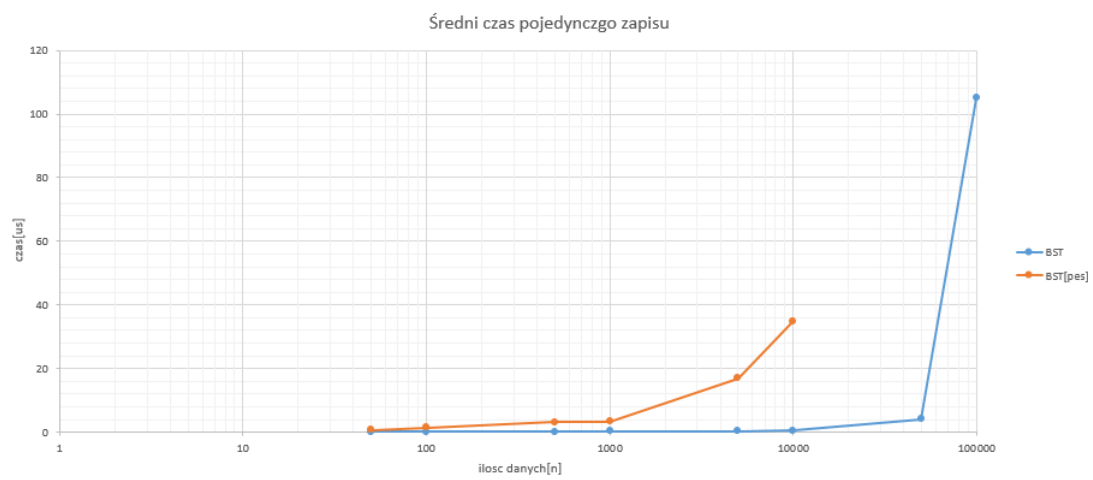
Rysunek 1: Wyniki odczytu pojedynczego elementu dla drzewa AVL



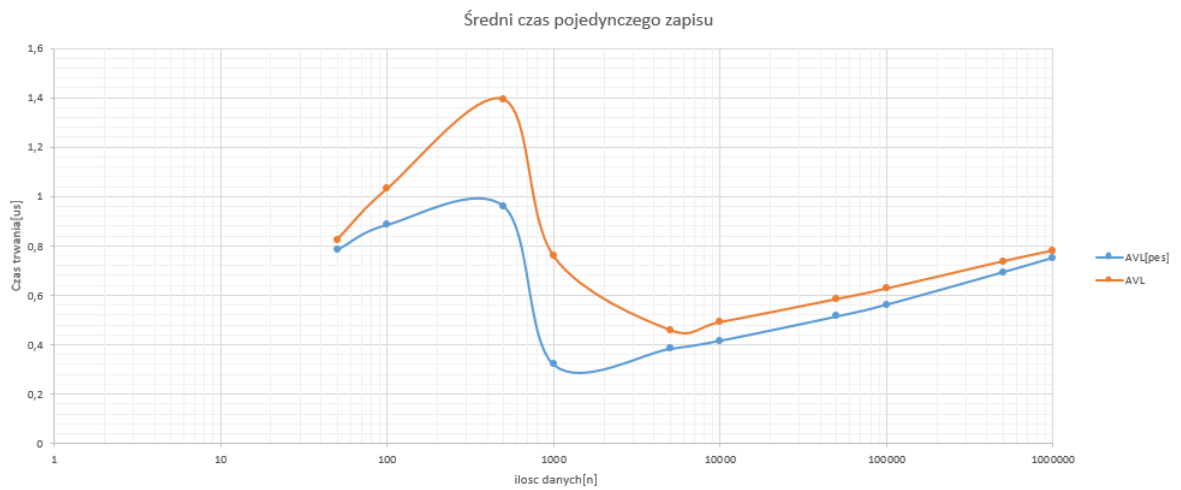
Rysunek 2: Wyniki odcztu pojedynczego elementu dla drzewa BST

### 3.2 Zapis

Ilość danych[n]	AVL pes[us]	AVL[us]	BST[us]	BST pes[us]
50	0,784	0,826	0,128	0,762
100	0,885	1,032	0,126	1,405
500	0,960	1,393	0,225	3,140
1000	0,320	0,758	0,296	3,363
5000	0,384	0,460	0,412	17,07
10000	0,417	0,492	0,58	35,010
50000	0,515	0,585	4,121	-
100000	0,563	0,628	105,0	-
500000	0,694	0,738	-	-
1000000	0,753	0,782	-	-



Rysunek 3: Wyniki zapisu pojedynczego elementu dla drzewa BST



Rysunek 4: Wyniki zapisu pojedynczego elementu dla drzewa AVL

## 4 Wnioski

- Zastosowanie mechanizmów balansujących drzewo AVL przyniosło zamierzany efekt. Przy podaniu na wejście programu danych posortowanych rosnąco, drzewo nie degeneruje się do postaci listy. Złożoność operacji odczytu i zapisu dla drzewa AVL są zatem zgodne z teoretyczną i wynosi  $O(\log_2 n)$
- W przypadku podania danych losowych na wejście dla drzewa AVL, prawidłową złożoność posiada operacja zapisu danych. Natomiast w przypadku operacji odczytu danych, otrzymane wyniki nie są zgodne z teoretycznymi. Sprawa jest o tyle dziwna, że monitorowanie pracy programu dla tego przypadku wykazało, że wyszukiwanie danych wartości na podstawie klucza odbywa się prawidłowo. Mimo to otrzymane wyniki świadczą o tym, że program zbyt wcześnie zakończył swoją pracę.
- Podanie na wejście danych losowych dla drzewa binarnego, nie prawidłową złożoność posiada operacja zapisu, ponieważ dla ilości danych powyżej 10k, złożoność jest liniowa.
- W przypadku operacji odczytu dla drzewa binarnego, podanie na wejście danych losowych powoduje przypadek podobny jak dla drzewa AVL.



- Podanie na wejście programu danych posortowanych rosnąco dla drzewa binarnego powoduje, degenerację drzewa do postaci listy.