

## Sprawozdanie

Autorzy: Angelika Ostrowska, Kacper Straszak

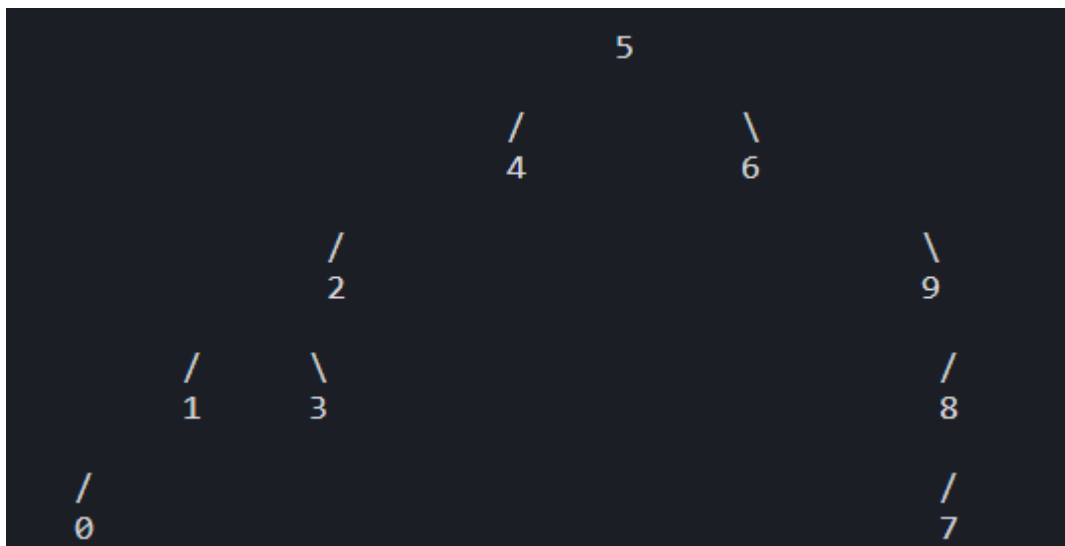
Data: 11.04.2023

AISDI Zad3

### Drzewa BST:

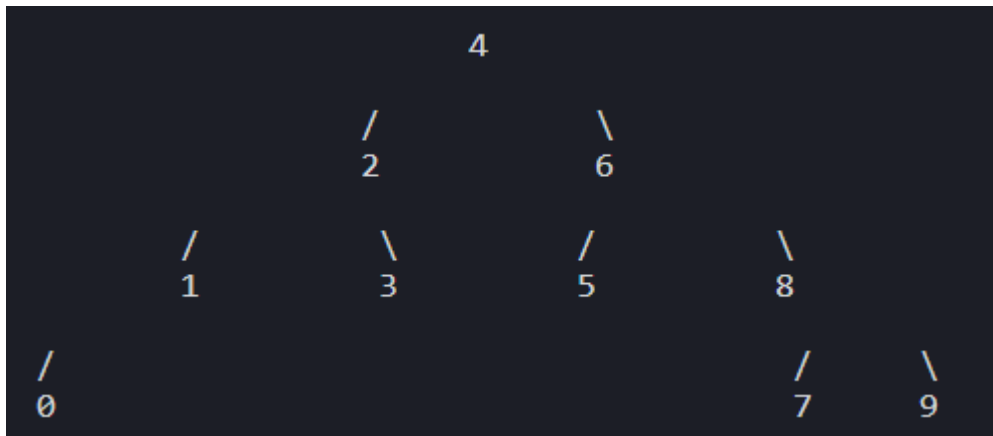
Drzewa poszukiwań binarnych to dynamiczne struktury danych dobre do szybkiego przeszukiwania w celu szybkiego znalezienia elementów. Jeśli drzewo zawierające 1000000 elementów jest korzystnie ułożone, to wymaga jedynie co najwyżej 20 porównań. Drzewo BST zbudowane jest z węzłów, w których przechowywane są dane. Węzły powiązane są ze sobą. Pierwszy z nich nazywa się korzeniem, a dwa kolejne z niego wychodzące nazywa się jego synami. Po lewo od korzenia (ojca) znajduje się syn o wartości od niego mniejszej, a z prawej o wartości większej.

Poniżej prezentacja drzewa BST dla wartości: 5,4,2,1,6,3,9,0,8,7:



### Drzewo AVL

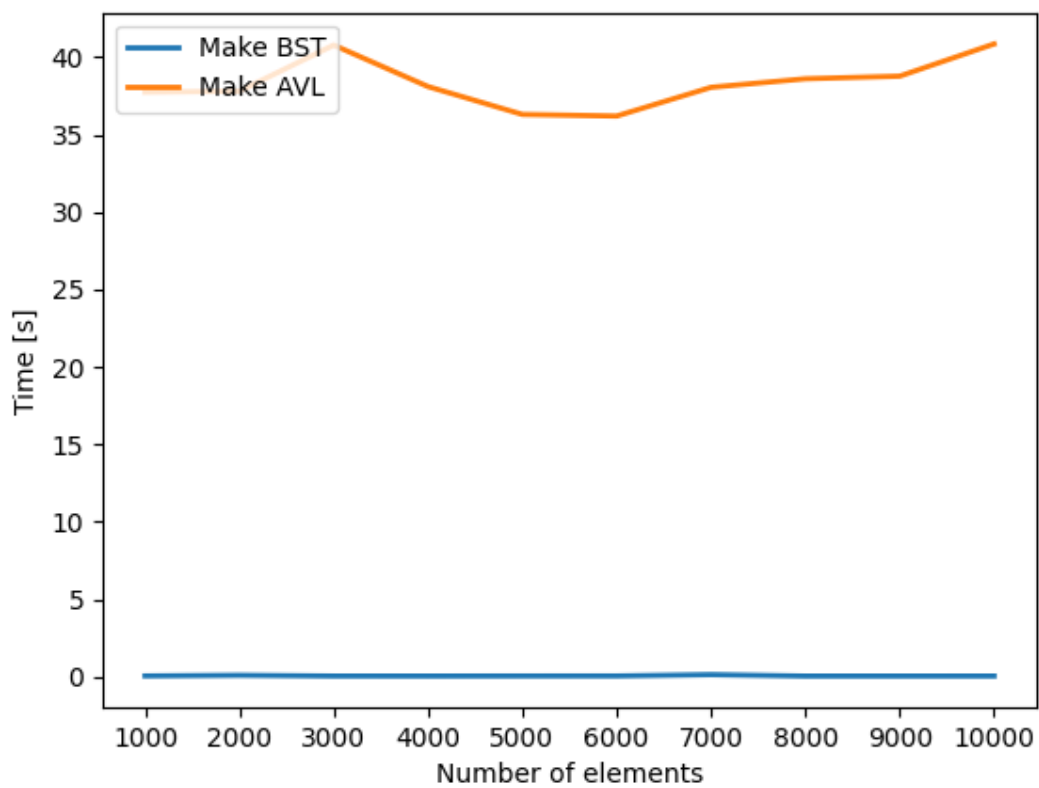
Drzewo AVL jest drzewem BST, które automatycznie równoważy swoje rozłożenie tak, by dla każdego węzła wysokość jego poddrzew dla obu synów różniła się maksymalnie o jeden. Drzewo równoważy się poprzez odpowiednie rotacje w prawo lub w lewo w kierunku korzenia. Istnieją cztery typy przełożeń: RR, LL, RL, LR, przy czym R i L określają sposób połączenia węzłów przed wykonaniem rotacji, np. LL - węzeł łączy się tylko lewymi krawędziami. Poniżej prezentacja drzewa BST dla wartości: 5,4,2,1,6,3,9,0,8,7:



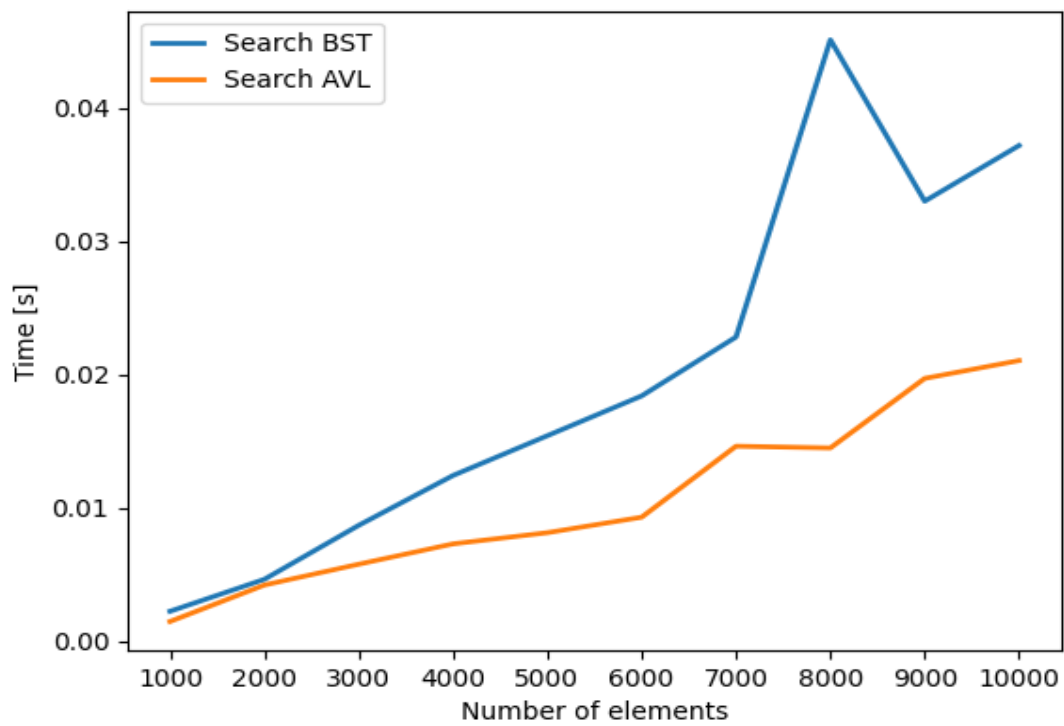
### Porównanie wyników

Porównanie wyników dla tworzenia drzew, wyszukiwania ich składników i usuwania elementów.

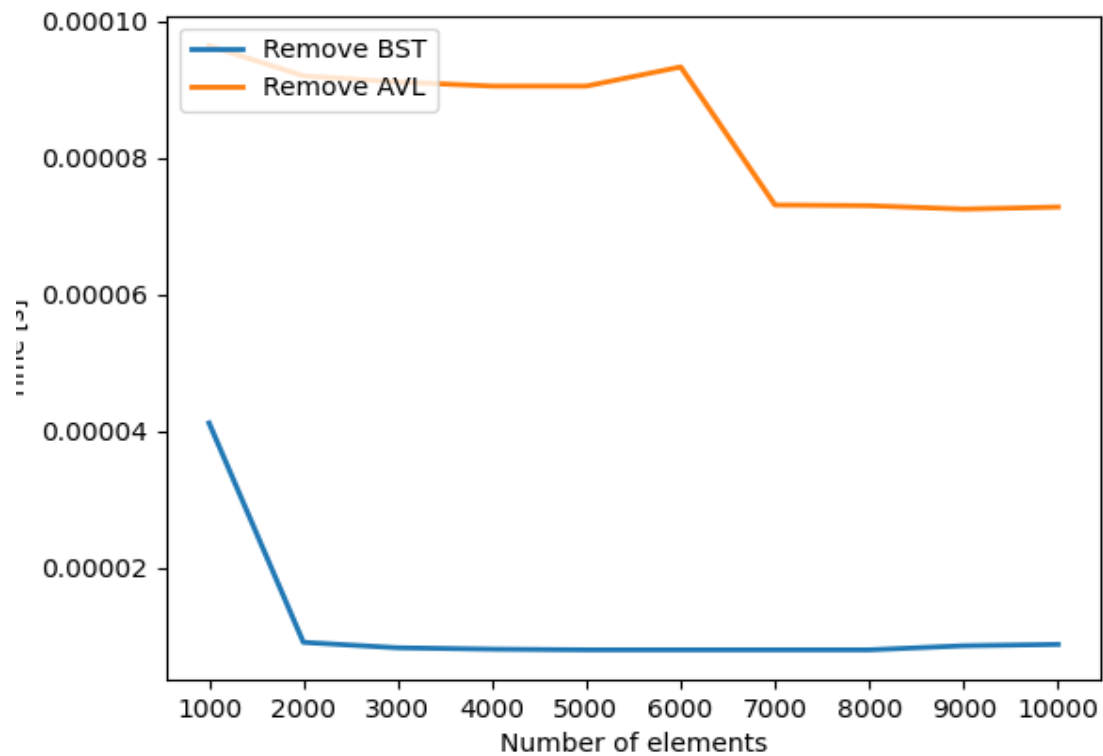
Tworzenie:



Szukanie:



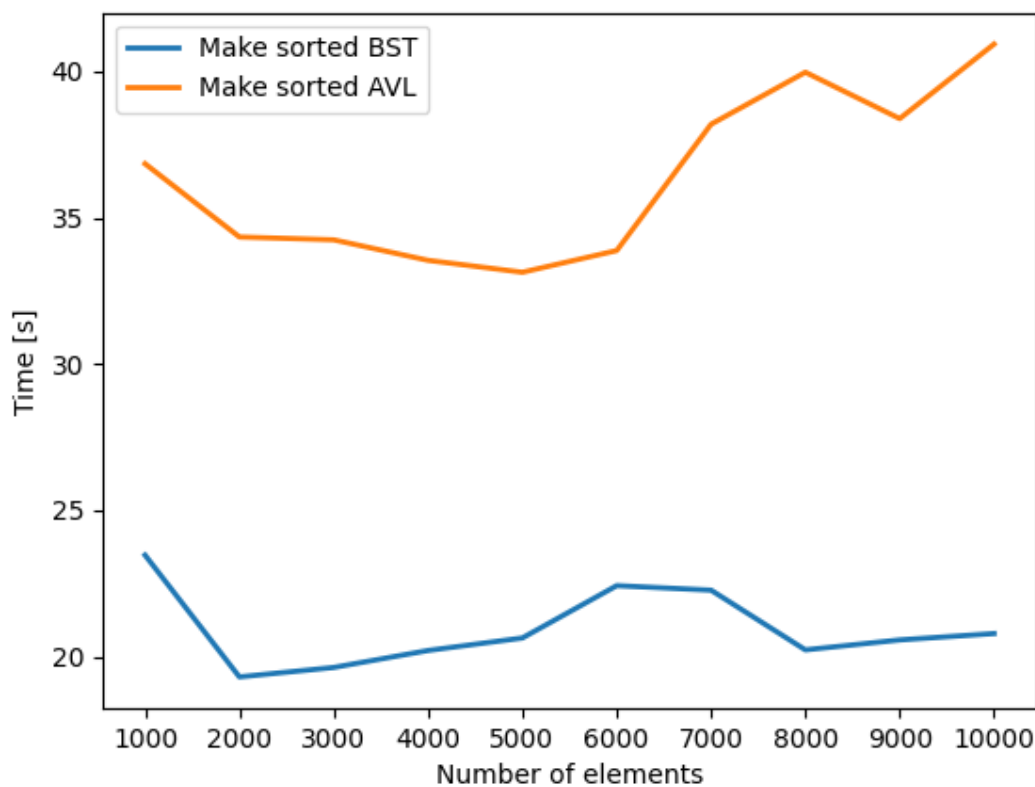
Usuwanie:



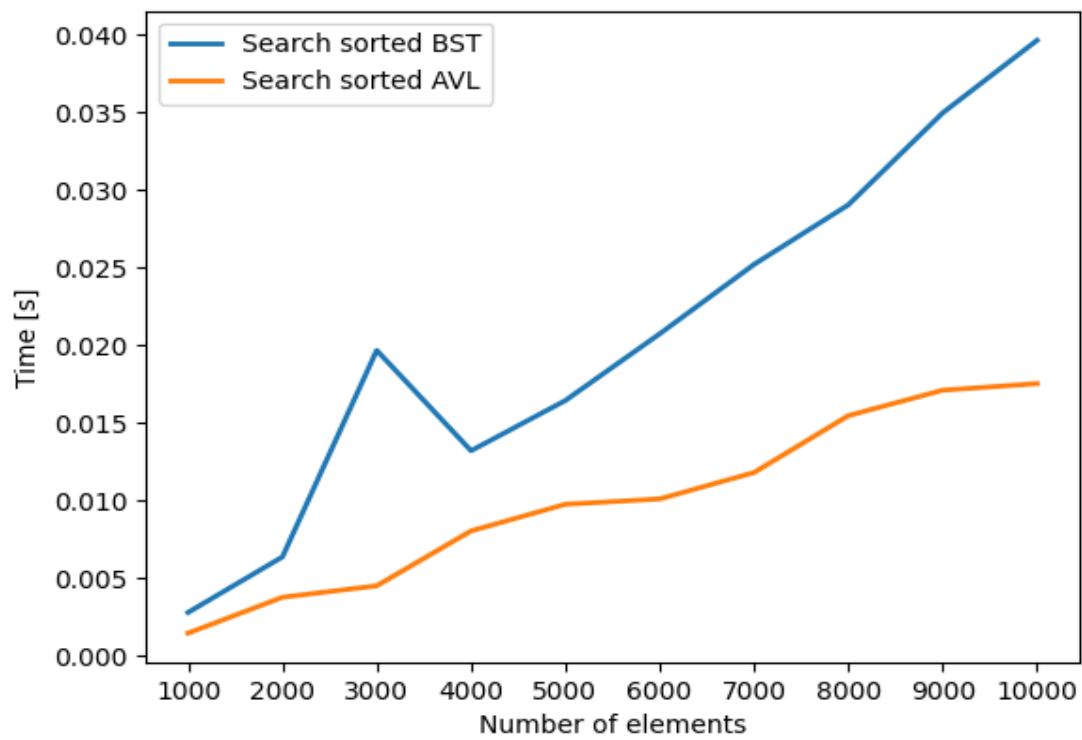
Drzewo BST tworzy się o wiele szybciej, ponieważ nie wymaga sprawdzania rozłożenia elementów i nie zamienia ich kolejności po sprawdzeniu. Drzewo AVL jest natomiast szybsze podczas przeszukiwania dużej ilości elementów. Szukanie w tym drzewie nie wymaga tak wielu przejść/sprawdzeń. Gdyby elementy były poukładane losowo (były wyszukiwane w takiej kolejności w jakiej budowane było z nich drzewo) to wyszukiwanie dużej liczby danych byłoby jeszcze szybsze. W szybkości usuwania elementów znowu wygrywa drzewo BST, ponownie problemem w drzewie AVL jest tu sprawdzanie czy rozłożenie węzłów jest optymalne i następująca rotacja przy nieoptymalnych przypadkach.

Porównanie drzew kiedy elementy drzewa są posortowane:

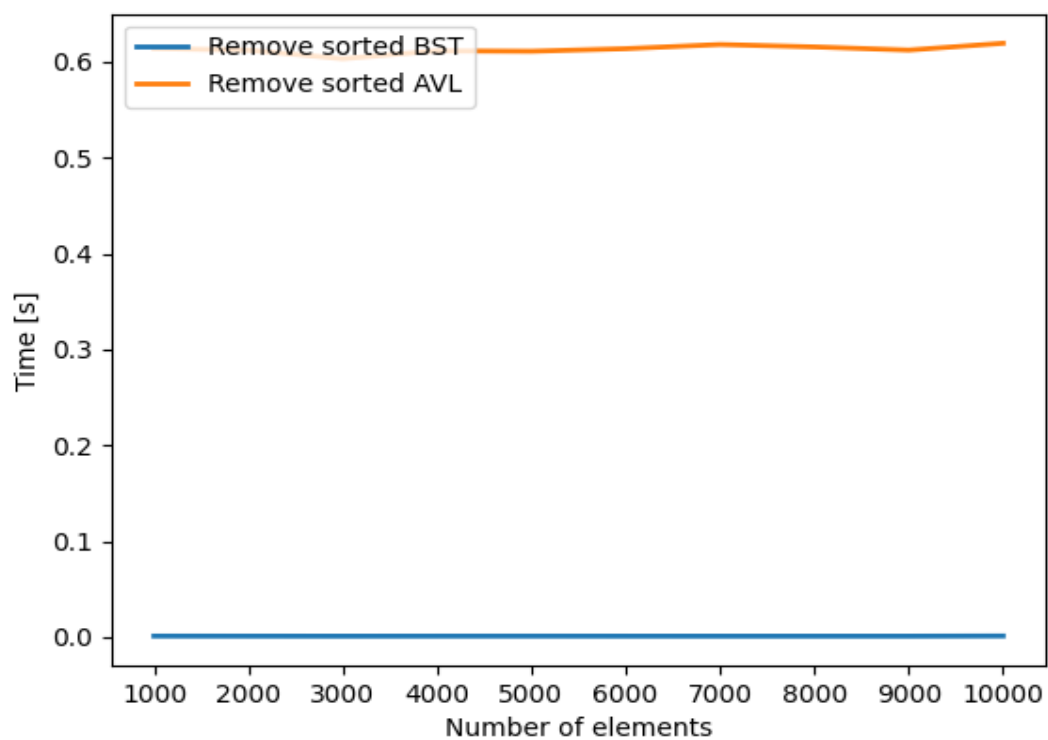
Tworzenie:



Szukanie:



Usuwanie:



Tworzenie i usuwanie danych posortowanych nie różni się mocno od wykonywania operacji z danymi podanymi losowo, natomiast szukanie danych nie w takiej kolejności w jakiej było tworzone drzewo wywołuje dłuższy czas poszukiwania i większe skoki czasowe podczas wyszukiwania dużej liczby elementów. Wyszukiwanie w drzewach BST jest także o wiele mniej regularne niż w wyregulowanym drzewie AVL.