ZŁOŻONE STRUKTURY DANYCH

Sprawozdanie

Aleksandra Jankowska,

Aleksandra Baumgart

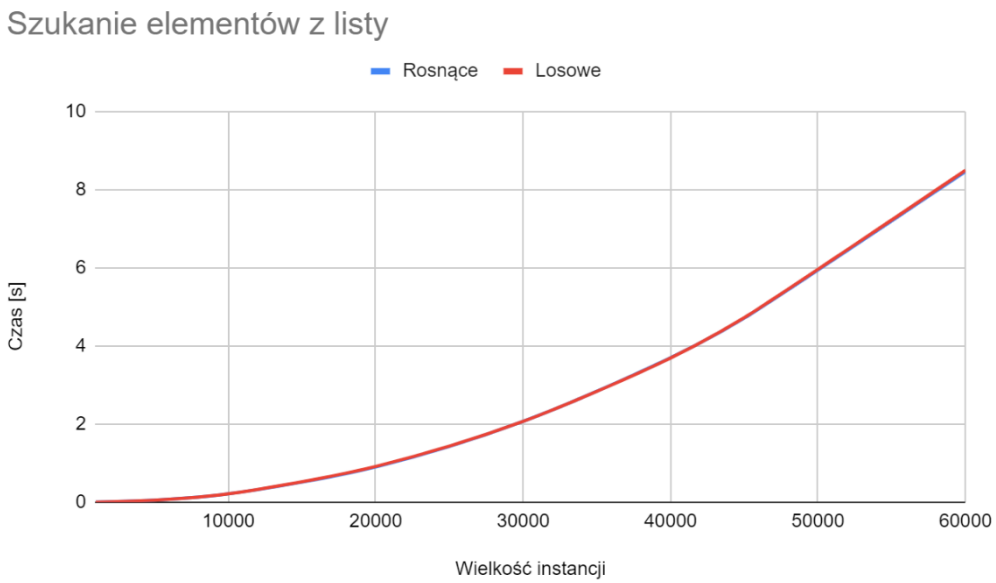
Lista jednokierunkowa – dodawanie elementów

	Czas dodawania elementów w zależności od algorytmu	
Wielkość instancji	Rosnące	Losowe
1000	0	0
5000	0	0
10000	0,001	0,001
20000	0,002	0,002
30000	0,004	0,003
40000	0,004	0,005
45000	0,005	0,005
60000	0,006	0,007

Wnioski: Różnice są niewielkie, dla niektórych wartości instancji praktycznie identyczne, odrobinę szybciej poradził sobie algorytm używający rozkładu rosnącego.

Lista jednokierunkowa – szukanie elementów

	Czas szukania elementów w zależności od algorytmu	
Wielkość instancji	Rosnące	Losowe
1000	0,003	0,002
5000	0,053	0,053
10000	0,214	0,214
20000	0,902	0,917
30000	2,07	2,071
40000	3,699	3,691
45000	4,712	4,724
60000	8,462	8,495



Wnioski: Znowu, różnice praktycznie niewidoczne, w tym przypadku trudno orzec, który algorytm zadziałał szybciej, dla kilku instancji wyniki pokryły się.

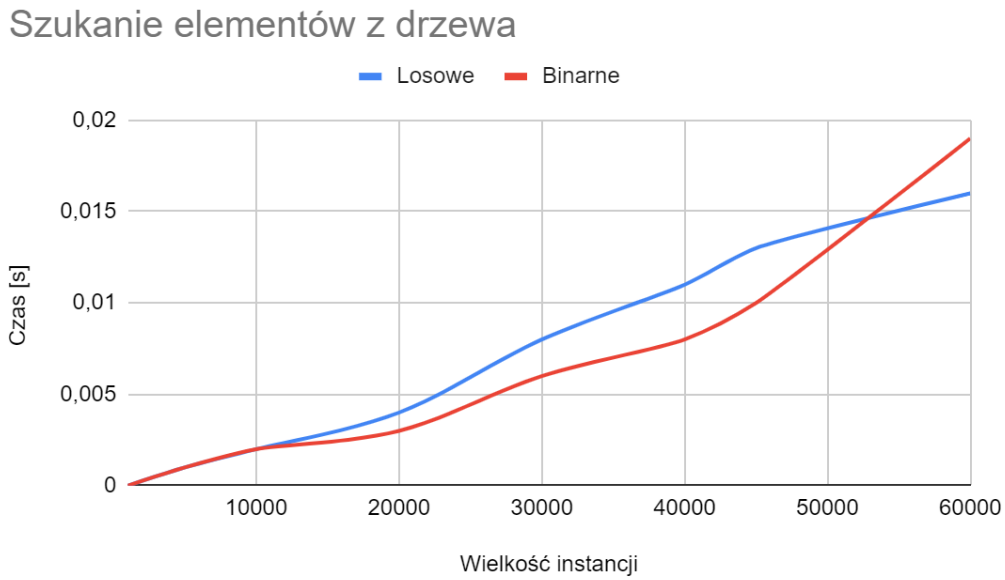
Drzewo BST – dodawanie elementów

	Czas dodawania elementów w zależności od algorytmu		
Wielkość instancji	Rosnące	Losowe	Binarne
1000	0,002	0,001	0,001
5000	0,086	0,002	0
10000	0,354	0,002	0,002
20000	1,446	0,006	0,003
30000	3,281	0,009	0,005
40000	5,825	0,013	0,007
45000	7,738	0,015	0,008
60000	14,85	0,028	0,014

Wnioski: Najszybciej działa przy rozkładzie losowym i binarnym elementów, o wiele dłużej przy rosnącym rozkładzie elementów. Najszybciej algorytm poradził sobie dla rozkładu binarnego.

Drzewo BST – szukanie elementów

	Czas szukania elementów w zależności od algorytmu		
Wielkość instancji	Rosnące	Losowe	Binarne
1000	0,003	0	0
5000	0,87	0,001	0,001
10000	0,355	0,002	0,002
20000	1,429	0,004	0,003
30000	3,289	0,008	0,006
40000	6,276	0,011	0,008
45000	7,72	0,013	0,01
60000	14,381	0,016	0,019



Wnioski: Znowu, najszybciej algorytm zadziałał przy rozkładzie losowym i binarnym elementów, a o wiele wolniej przy rosnącym rozkładzie elementów. Dla większości instancji najszybciej zadziałał przy rozkładzie binarnym elementów.

Wnioski

Dodawanie elementów do drzewa oraz szukanie elementów z drzewa w sposób rosnący jest najmniej wydajne czasowo. Z powodu dużej różnicy skali zostały one przedstawione na oddzielnych wykresach w porównaniu do losowych i binarnych.

W przypadku drzew zdecydowanie bardziej widać różnice w działaniach algorytmów i tutaj wygrywa działanie dla rozkładu binarnego, podczas gdy jeśli chodzi o listy te różnice są minimalne – algorytm jest praktycznie identycznie wydajny dla obu rozkładów.