

Sebastian Sobkowiak

Algorytmy i struktury danych

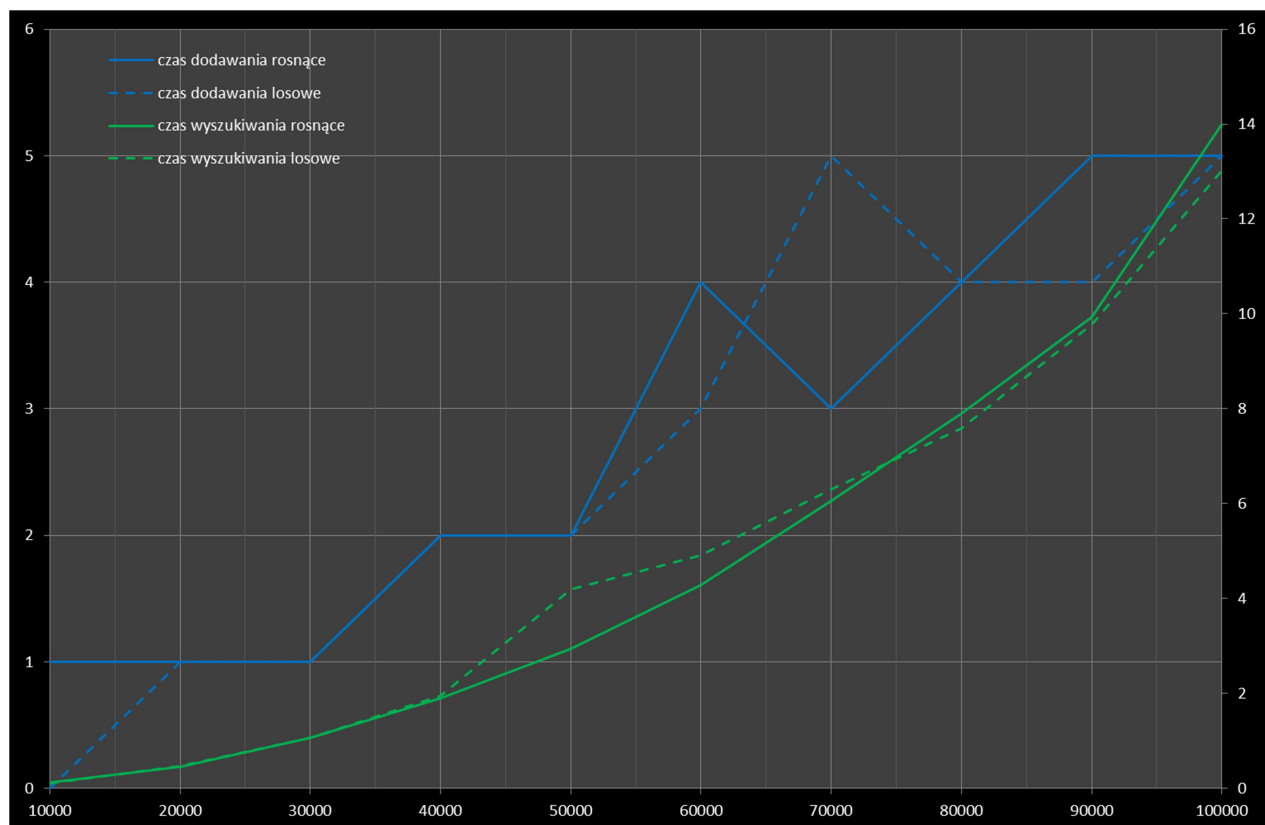
Projekt 2 – lista i BST

Lista (C):

czas dodawania jest w ms

czas wyszukiwania jest w s

	czas dodawania rosn	czas dodawania loso	czas wyszukiwania rd	czas wyszukiwania lo
10000	1	0	0,115	0,113
20000	1	1	0,461	0,463
30000	1	1	1,055	1,066
40000	2	2	1,905	1,943
50000	2	2	2,951	4,188
60000	4	3	4,274	4,911
70000	3	5	6,052	6,296
80000	4	4	7,909	7,587
90000	5	4	9,935	9,789
100000	5	5	13,984	13,018

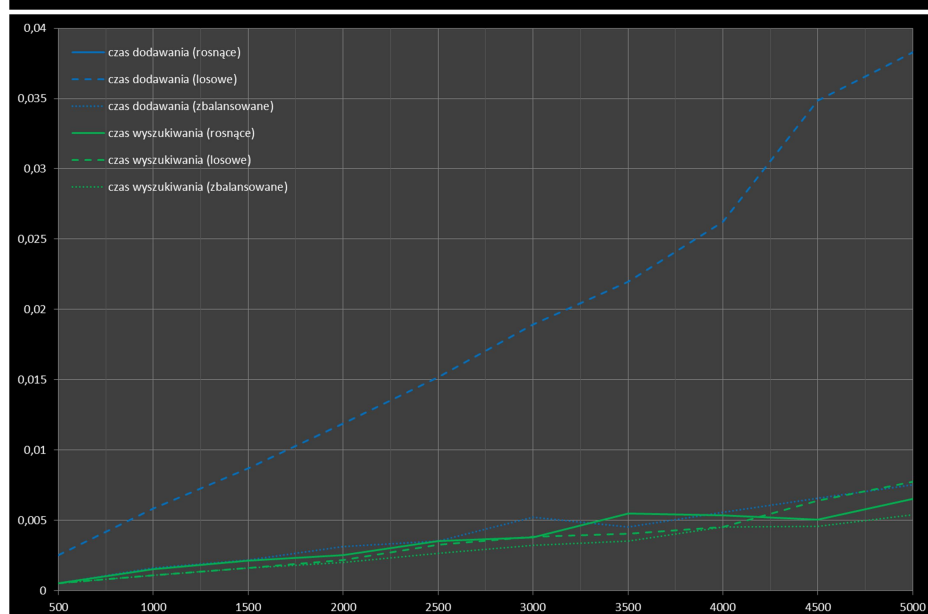
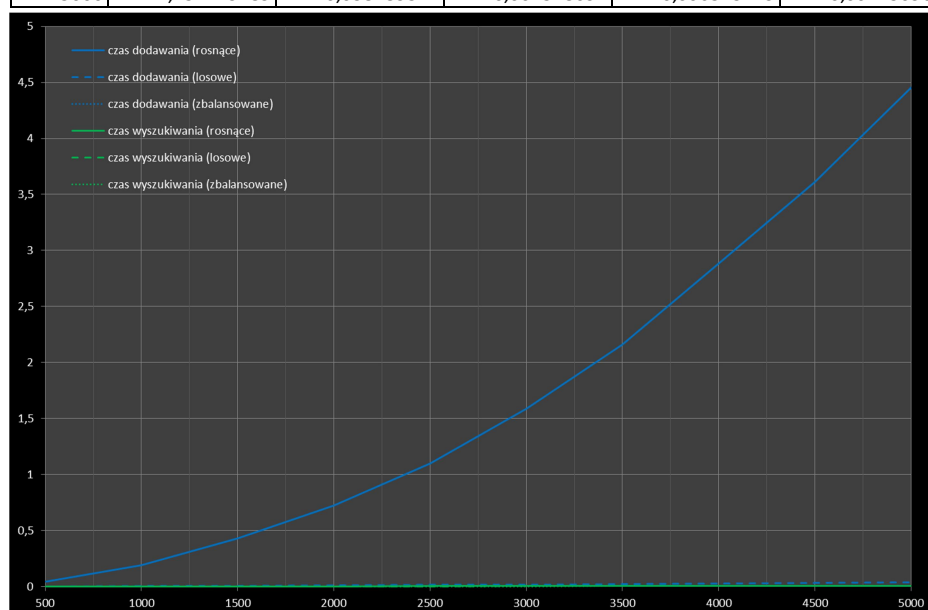


Dodawanie i wyszukiwanie elementów dla listy ma złożoność czasową wykładniczą.

BST (Py):

Próba nr 1, wielkości instancji [500,1000,1500,2000,2500,3000,3500,4000,4500,5000]

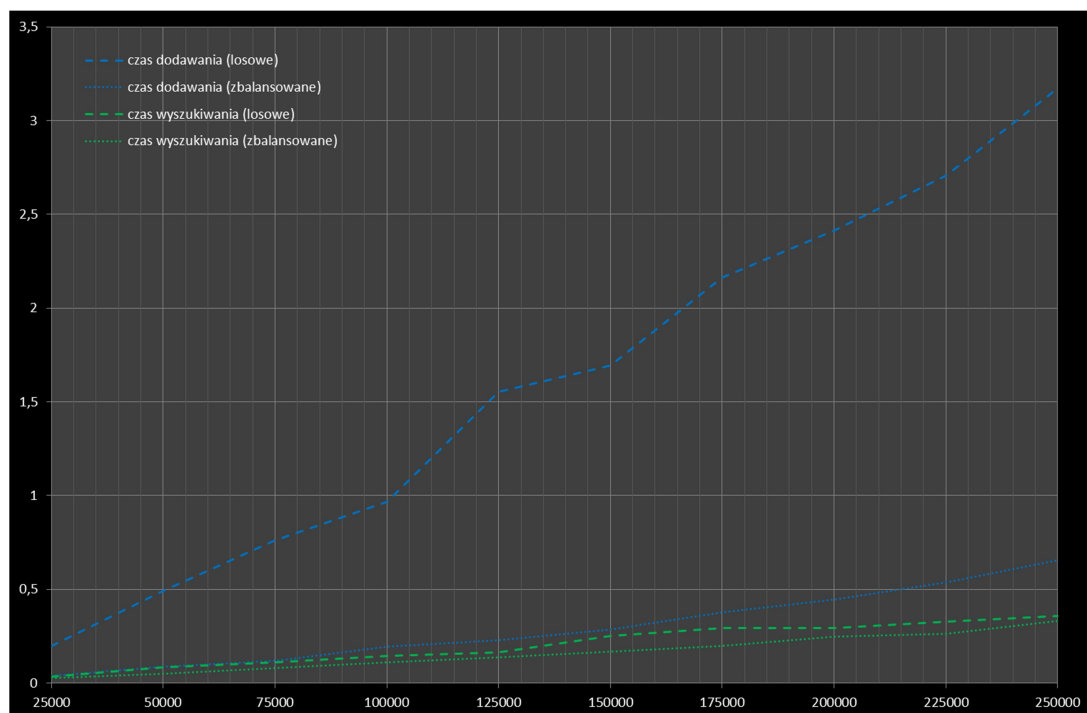
wielkość instancji	czas dodawania (rosnące)	czas dodawania (losowe)	czas dodawania (zbalansowane)	czas wyszukiwania (rosnące)	czas wyszukiwania (losowe)	czas wyszukiwania (zbalansowane)
500	0,045198679	0,002520561	0,000542164	0,000505924	0,000504732	0,000542402
1000	0,190397978	0,005836487	0,001629114	0,001514196	0,001084566	0,001084805
1500	0,430754423	0,00868845	0,002170563	0,002124071	0,001627684	0,001621485
2000	0,725895882	0,01189208	0,003126383	0,0025208	0,002169609	0,002016306
2500	1,10014534	0,015177011	0,003534794	0,003542423	0,003257751	0,002675772
3000	1,587951422	0,018948078	0,005221367	0,003773689	0,00381422	0,003208876
3500	2,160302639	0,021989584	0,004540682	0,005491018	0,004032135	0,003528833
4000	2,881044388	0,026230574	0,005558729	0,005348921	0,00453496	0,004537821
4500	3,613310575	0,034846306	0,006563902	0,005038261	0,006383657	0,004550695
5000	4,452249289	0,038289547	0,007548094	0,006513119	0,007730961	0,005385876



Dla danych rosnących drzewo nie jest wydajne, ponieważ wszystkie kolejne elementy trafiają zawsze do prawej gałęzi, a lewa zostaje pusta, przypomina to więc bardziej listę niż drzewo oraz powoduje, że drzewo ma mnóstwo poziomów, a rekurencyjne dodawanie elementów wydłuża się znacząco przez dłuższą drogę dotarcia do miejsca docelowego.

Próba nr 2, bez danych rosnących, wielkości instancji [25000,50000,75000,100000,125000,150000,175000,200000,225000,250000]

wielkość instancji	czas dodawania (losowe)	czas dodawania (zbalansowane)	czas wyszukiwania (losowe)	czas wyszukiwania (zbalansowane)
25000	0,196617365	0,038057089	0,034894466	0,02784729
50000	0,491012096	0,087449551	0,084424257	0,051462173
75000	0,760779858	0,119835377	0,111540079	0,081032515
100000	0,9677248	0,194954157	0,145820141	0,111323833
125000	1,554302931	0,229356527	0,164276123	0,138035297
150000	1,693052292	0,28718853	0,252331734	0,167155266
175000	2,162500858	0,37699914	0,295138121	0,197608948
200000	2,413128138	0,444301367	0,292817354	0,24613905
225000	2,706795692	0,538124323	0,328874826	0,261977196
250000	3,17193675	0,655878305	0,358842134	0,330761909



Dla różnych wielkości instancji czasy dodawania i wyszukiwania wykazują złożoność czasową liniową. Zbalansowane drzewa są podobnie wydajne w wyszukiwaniu elementów do losowo zapętnionych drzew, ale są o wiele bardziej wydajne w dodawaniu elementów.