

Martyna Demczuk  
Julia Bandera

Data oddania: 05.04.2020

# **Sprawozdanie**

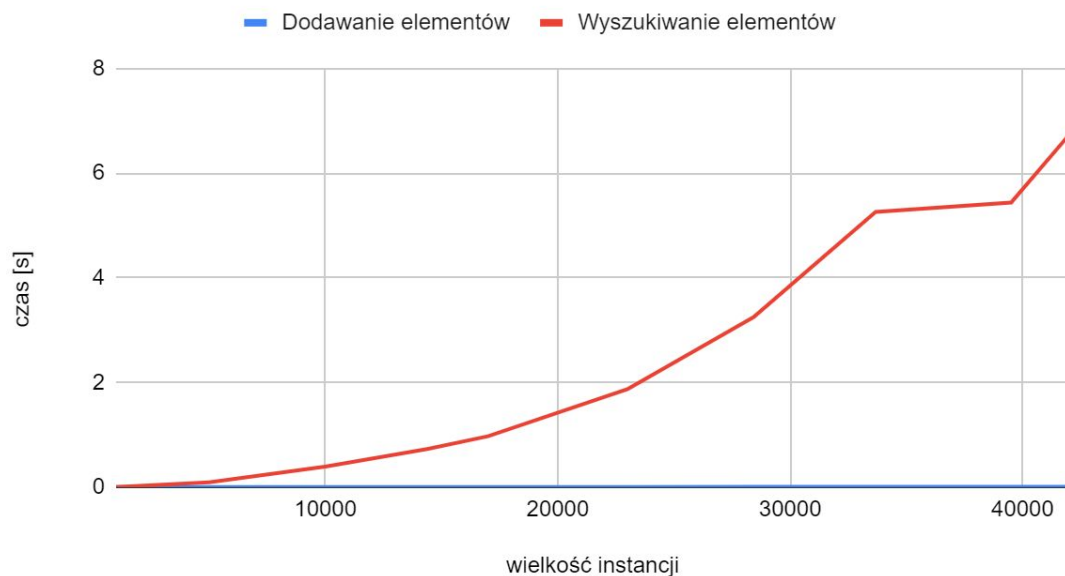
## **Lista jednokierunkowa i binary search tree**

Tabele i wykresy znajdujące się w sprawozdaniu przedstawiają czasy, które są potrzebne, by dodać oraz wyszukać wszystkie elementy na listę jednokierunkową bądź binary search tree. Dla listy jednokierunkowej są dwie metody dodawania elementów (rosnąco, losowo), a dla drzewa BST trzy (rosnąco, losowo, optymalnie).

## LISTA JEDNOKIERUNKOWA INCREASING

Wielkość instancji	Czas [s]	
	Dodawanie elementów	Wyszukiwanie elementów
1000	0,001	0,002
5000	0,001	0,094
10000	0,001	0,391
14389	0,003	0,728
17000	0,003	0,973
23000	0,003	1,872
28432	0,005	3,25
33675	0,005	5,259
39510	0,006	5,438
42000	0,006	6,734

Lista jednokierunkowa - Increasing

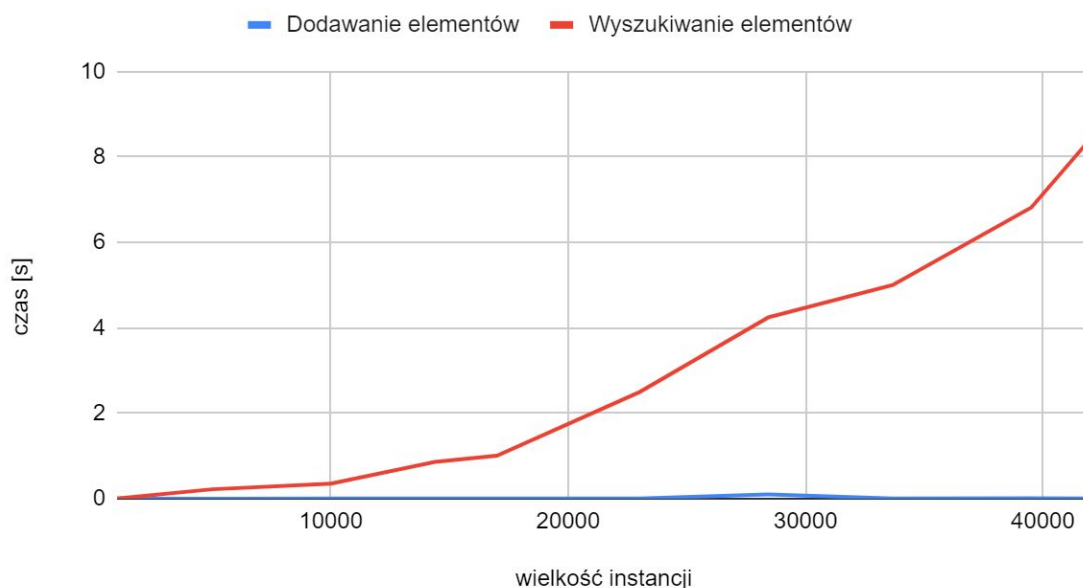


Przy wartościach dodawanych do listy rosnąco czas dodawania i wyszukiwania elementów rośnie wraz ze wzrostem ilości dodanych liczb. Najszybciej dodano 1000, 5000 i 10000 liczb (0,001 s), a najwolniej 39510 i 42000 liczb. Najszybciej zostały wyszukiwane wszystkie elementy przy 1000 liczb (0,002 s), a najwolniej przy 42000 (6,734 s).

## RANDOM

Wielkość instancji	Czas [s]	
	Dodawanie elementów	Wyszukiwanie elementów
1000	0,005	0,005
5000	0	0,219
10000	0,002	0,35
14389	0,002	0,862
17000	0,002	1,007
23000	0,005	2,492
28432	0,1	4,245
33675	0,006	5
39510	0,009	6,815
42000	0,006	8,423

### Lista jednokierunkowa - Random

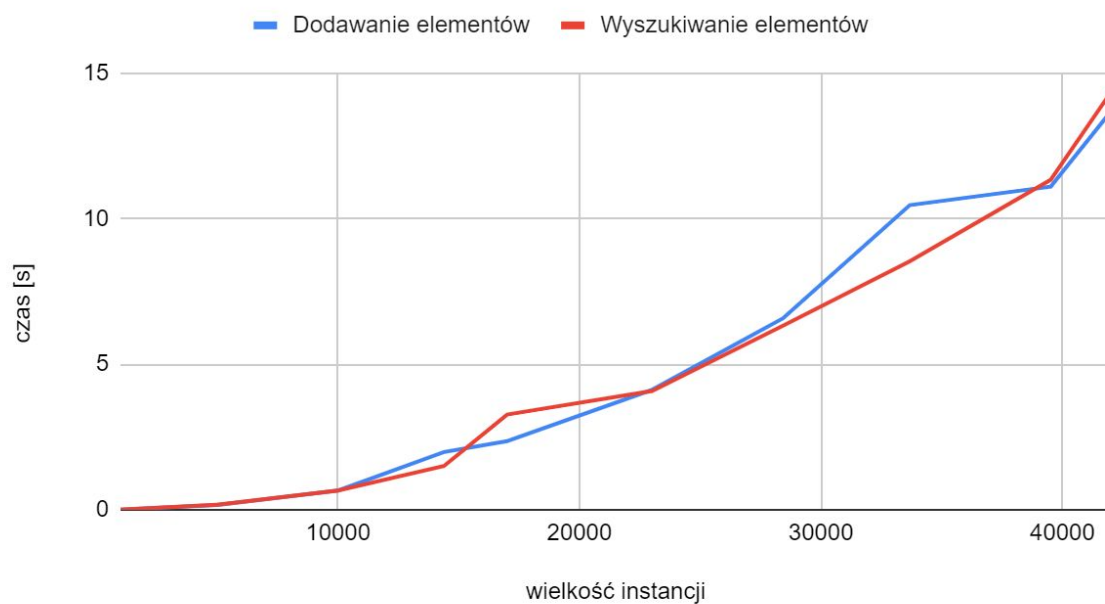


Przy wartościach dodawanych do listy w losowej kolejności czas dodawania elementów nie zależy od ilości liczb. Dodanie 42 tysięcy liczb trwało 0,006 sekundy, a 28432 liczb aż 0,1 sekundy (najdłużej ze wszystkich). Najkrócej trwało dodanie 5000 liczb. Czas wyszukiwania elementów rośnie wraz ze wzrostem ilości liczb na liście jednokierunkowej. Najszybciej wykonano wyszukiwanie wszystkich elementów przy 1000 liczb - 0,005 sekundy, a najwolniej przy 42000 liczb - 8,423 sekund.

## BINARY SEARCH TREE INCREASING

Wielkość instancji	Czas [s]	
	Dodawanie elementów	Wyszukiwanie elementów
1000	0,008	0,007
5000	0,162	0,174
10000	0,662	0,661
14389	1,983	1,503
17000	2,356	3,27
23000	4,12	4,081
28432	6,577	6,325
33675	10,463	8,53
39510	11,099	11,335
42000	13,672	14,332

BST - Increasing

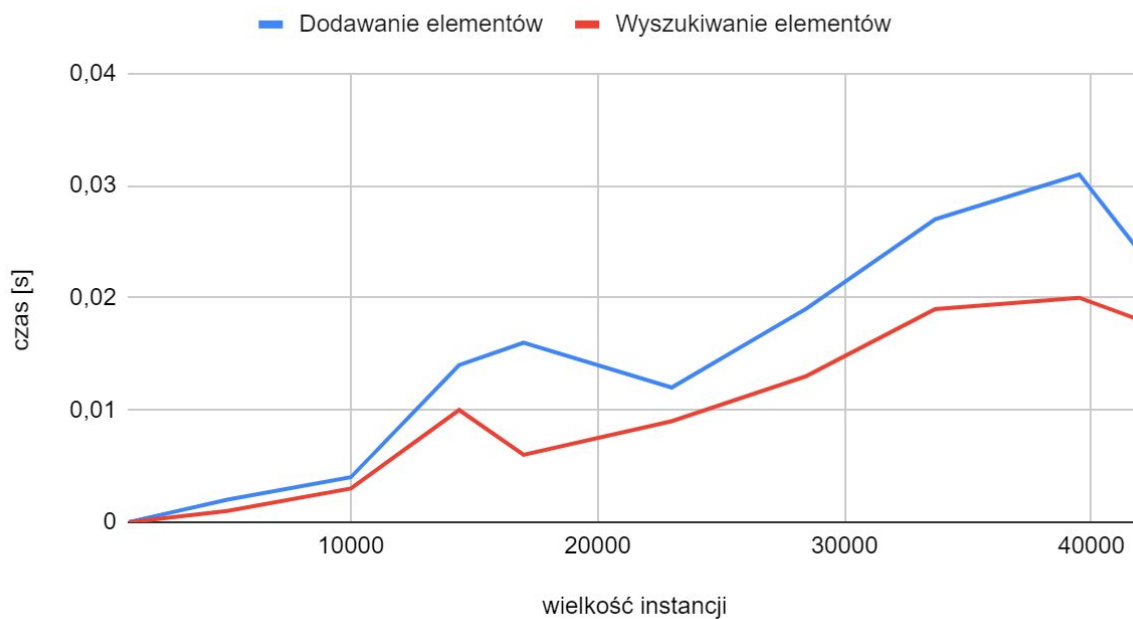


Wraz ze wzrostem wielkości instancji rośnie czas dodawania i wyszukiwania wszystkich elementów w drzewie BST. Najkrócej trwało to przy 1000 liczb - dodanie elementów zajęło 0,008 sekundy, a wyszukanie 0,007 sekundy. Najdłużej przy 42000 liczb - dodanie zajęło 13,672 sekund, a wyszukanie 14,332 sekund.

## RANDOM

Wielkość instancji	Czas [s]	
	Dodawanie elementów	Wyszukiwanie elementów
1000	0	0
5000	0,002	0,001
10000	0,004	0,003
14389	0,014	0,01
17000	0,016	0,006
23000	0,012	0,009
28432	0,019	0,013
33675	0,027	0,019
39510	0,031	0,02
42000	0,024	0,018

### BST - Random

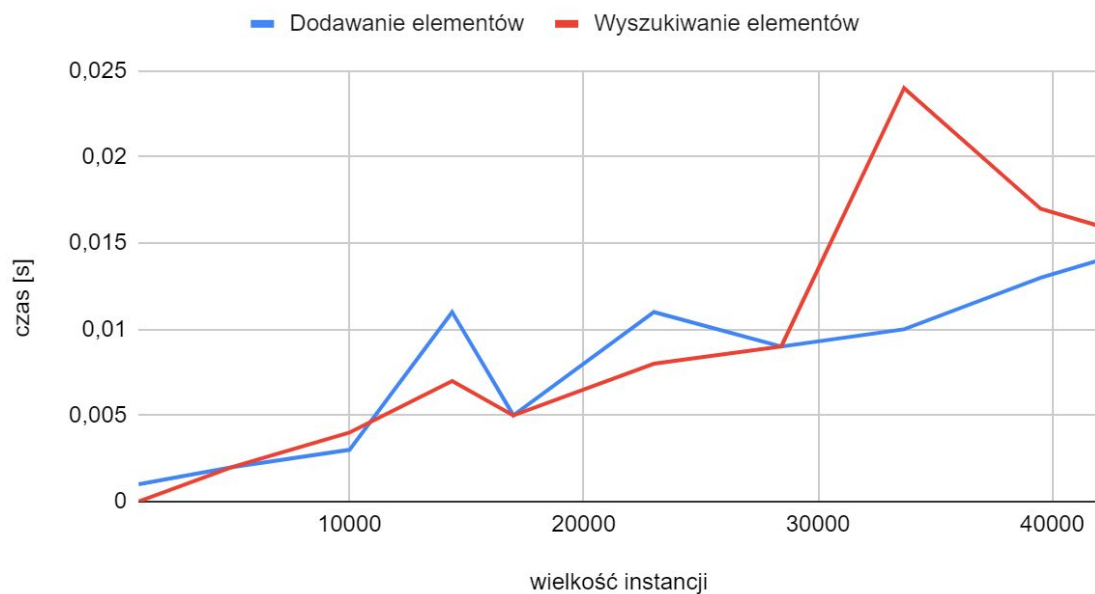


Przy dodaniu elementów do drzewa BST w kolejności losowej czas dodania i wyszukania wszystkich elementów nie zależy od wielkości instancji. Dla 1000 liczb dodanie i wyszukanie elementów trwało najkrócej. Najdłużej dodawanie i wyszukiwanie trwało przy 39510 liczbach (dodawanie - 0,031 s, wyszukiwanie - 0,02 s).

## BINARY

Wielkość instancji	Czas [s]	
	Dodawanie elementów	Wyszukiwanie elementów
1000	0,001	0
5000	0,002	0,002
10000	0,003	0,004
14389	0,011	0,007
17000	0,005	0,005
23000	0,011	0,008
28432	0,009	0,009
33675	0,01	0,024
39510	0,013	0,017
42000	0,014	0,016

### BST - Binary



Przy dodawaniu liczb do drzewa BST w sposób optymalny czas dodawania i wyszukiwania wszystkich elementów nie zależy od wielkości instancji. Przy 1000 liczb dodawanie trwało 0,001 sekundy a wyszukiwanie 0 sekund (najszybciej ze wszystkich). Najwolniej dodane zostało 42000 liczb (0,014 s). Wyszukiwanie wszystkich elementów najdłużej trwało przy 33675 liczbach (0,024 s).

## WNIOSKI

Czas dodawania liczb rosnąco wygenerowanych do listy jednokierunkowej jest nieporównywalnie szybszy w porównaniu do drzewa BST. Również wyszukiwanie tych elementów odbywa się szybciej w liście jednokierunkowej.

Z kolei w przypadku liczb generowanych losowo w liście jednokierunkowej tylko dodawanie wykonuje się szybciej niż w drzewie BST, natomiast wyszukiwanie tych elementów zachodzi już znacznie szybciej w drzewie BST niż w liście.

Liczby wygenerowane w sposób optymalny były brane pod uwagę tylko w przypadku drzewa BST, dlatego nie można go porównać pod tym względem z listą jednokierunkową. Jest możliwe natomiast porównanie czasów dodawania elementów do drzewa i ich wyszukiwania: nie różnią się one znacząco z wyjątkiem przypadku jednej instancji.

Najszybsze dodawanie elementów miało miejsce w przypadku dodawania liczb rosnąco do listy jednokierunkowej, natomiast najwolniejsze w przypadku dodawania liczb rosnąco do drzewa BST.

Najszybsze wyszukiwanie elementów odbyło się w przypadku wyszukiwania liczb generowanych w sposób optymalny w drzewie BST, natomiast najwolniejsze w przypadku wyszukiwania liczb wygenerowanych rosnąco w drzewie BST.