

Politechnika Wrocławska Wydział Informatyki i
Telekomunikacji

Sprawozdanie

**SDiZO – badanie efektywności operacji na danych w
podstawowych strukturach danych**

Autor: Stanisław Strauchold 259142

Grupa: Piątek TP 11:15

Prowadzący: dr inż. Dariusz Banasiak

Termin oddania: 22.04.2022

1. Wstęp

Zadaniem była implementacja wybranych struktur danych oraz przeprowadzenie na nich testów w celu sprawdzenia złożoności wykonywanych na nich operacji.

Zaimplementowane struktury:

- tablica
- lista dwukierunkowa
- kopiec binarny

2. Opis zaimplementowanych struktur

2.1. Tablica

Tablica była alokowana dynamicznie, tzn. zawsze zajmowała dokładnie tyle pamięci, ile danych przechowywała. Z racji na konieczność relokowania tablicy przy każdym dodaniu i usunięciu elementu średnia oraz pesymistyczna złożoność obliczeniowa tych operacji wynosiła $O(n)$. Taką samą złożoność posiada operacja wyszukiwania elementu w strukturze.

2.2. Lista dwukierunkowa

Tablica została zaimplementowana z wykorzystaniem struktury, która reprezentowała pojedynczy element listy. Każda struktura przechowywała klucz oraz wskaźnik na element poprzedni i następny. Oprócz tego wyróżnione były głowa i ogon listy, a co za tym idzie operacje dodawania i odejmowania elementów dla początku i końca listy mają złożoność średnią i pesymistyczną $O(1)$. Operacje dodawania i odejmowania dla elementów wewnątrz listy oraz operacja wyszukiwania mają złożoność średnią i pesymistyczną $O(n)$. Wynika to z faktu, iż chcąc się do nich dostać musimy przechodzić po kolei przez elementy listy.

2.3. Kopiec binarny

Kopiec został zaimplementowany z wykorzystaniem tablicy, w której przechowywane były elementy struktury. Operacje dodawania i odejmowania w kopcu mają złożoność średnią $O(\log(n))$ oraz pesymistyczną $O(n)$, natomiast operacja wyszukiwania ma średnią i pesymistyczną złożoność obliczeniową $O(n)$, z racji wykorzystania tablicy w celu implementacji struktury.

3. Przebieg procesu testowania

Dla każdej operacji w każdej strukturze proces testowania prezentował się w ten sam sposób. Generowany był losowy zestaw danych z zakresu $<-10000;10000>$, następnie wywoływana była metoda wykonująca daną operację i mierzony był czas jaki był potrzebny do wykonania metody. Pomiar przeprowadzany był dla różnych rozmiarów

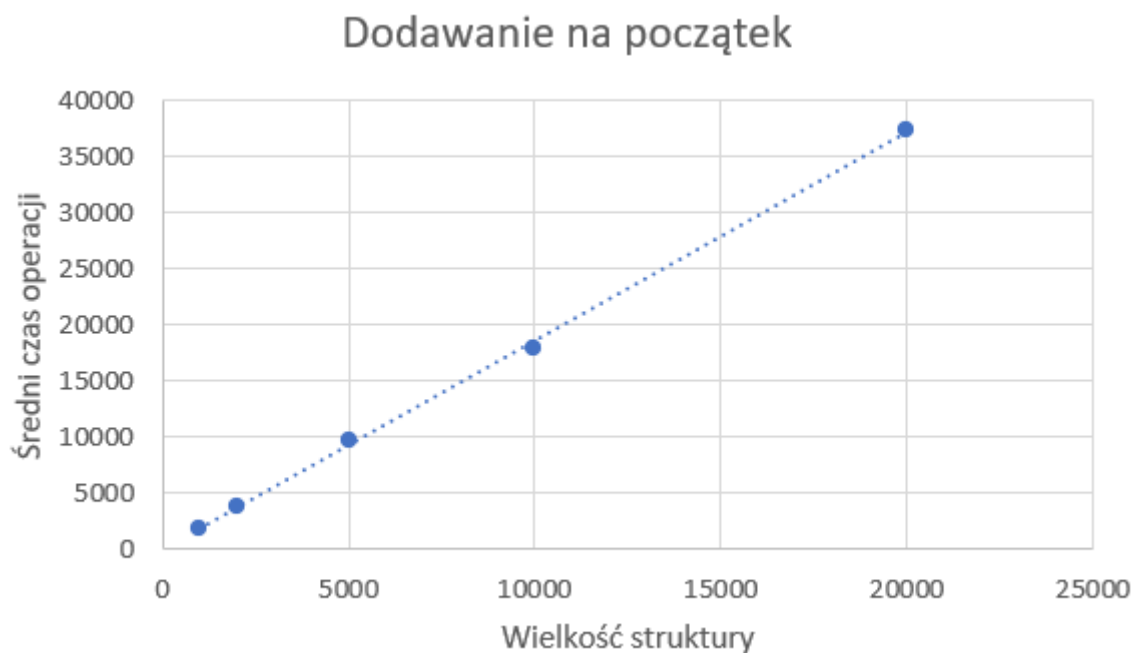
struktur: 1000, 2000, 5000, 10000 oraz 20000 elementów. Dla każdego z tych rozmiarów pomiar mierzony był 100 razy, za każdym razem dla innego zestawu danych. Następnie liczony był średni czas potrzebny do wykonania danej operacji.

4. Wyniki pomiarów

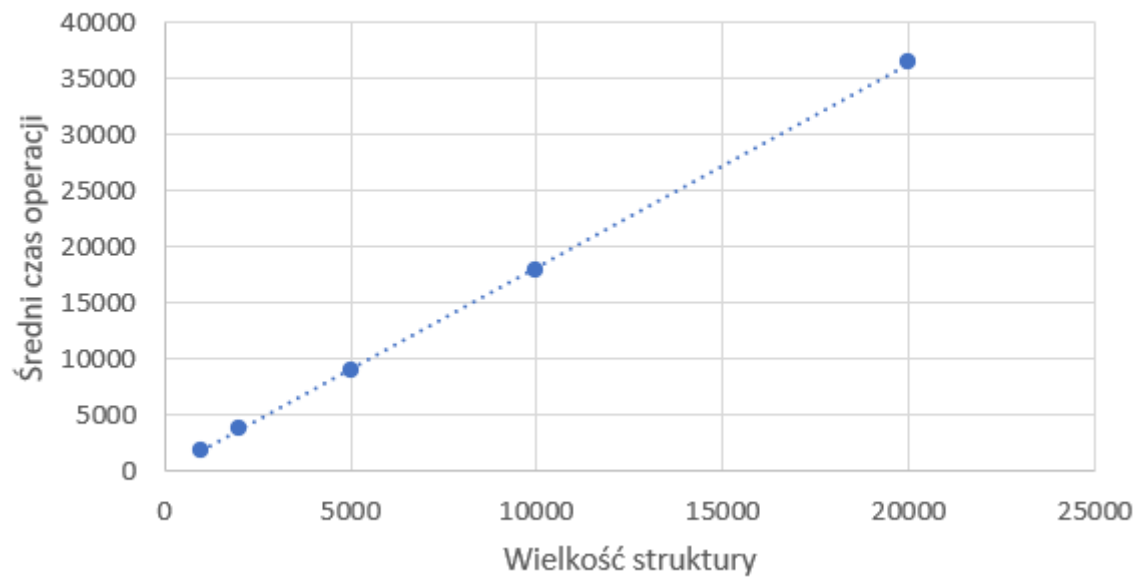
Czas wykonywanych operacji przedstawiony został w nanosekundach

4.1. Tablica

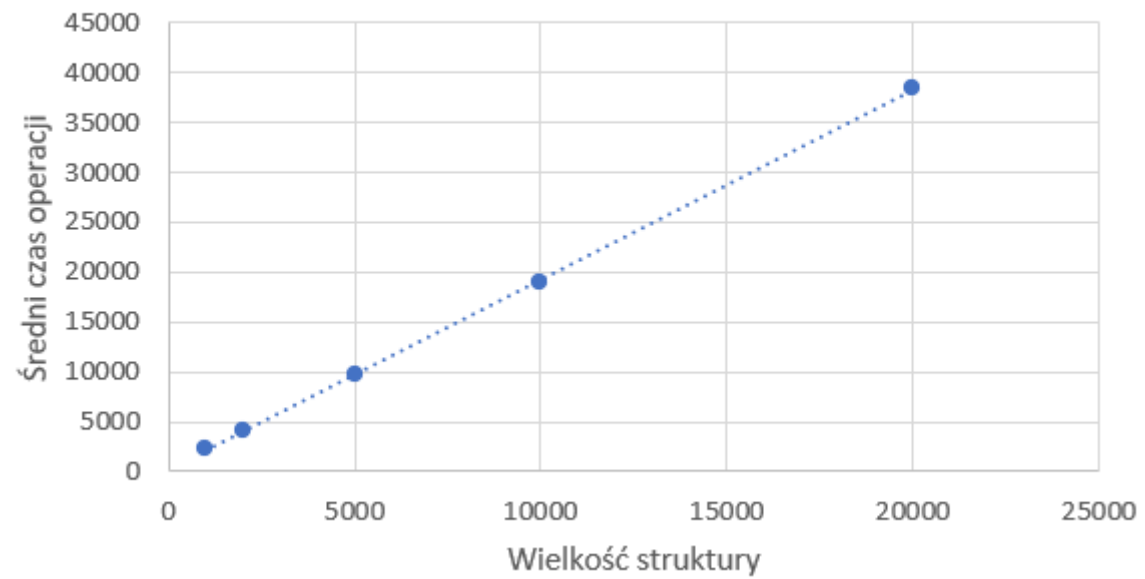
Operacja/Wielkość struktury	1000	2000	5000	10000	20000
DODAWANIE NA POZĄTEK	1918	3784	9671	18019	37 353
DODAWANIE NA KONIEC	1954	3908	9007	17870	36516
DODAWANIE NA WYBRANA POZYCJE	2210	4134	9769	19046	38424
USUWANIE PIERWSZEGO ELEMENTU	2018	3745	9323	17955	37922
USUWANIE OSTATNIEGO ELEMENTU	2118	4061	9924	19851	38532
USUWANIE WYBRANEGO ELEMENTU	2295	4001	9368	19139	39251
WYSZUKIWANIE ELEMENTU	1595	2793	6249	8364	14168



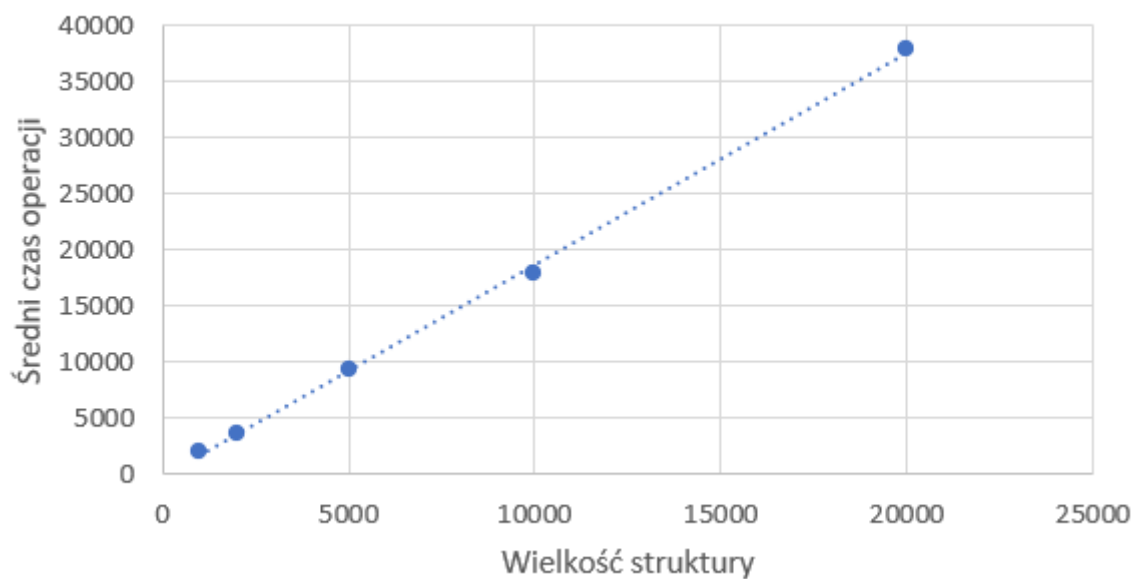
Dodawanie na koniec



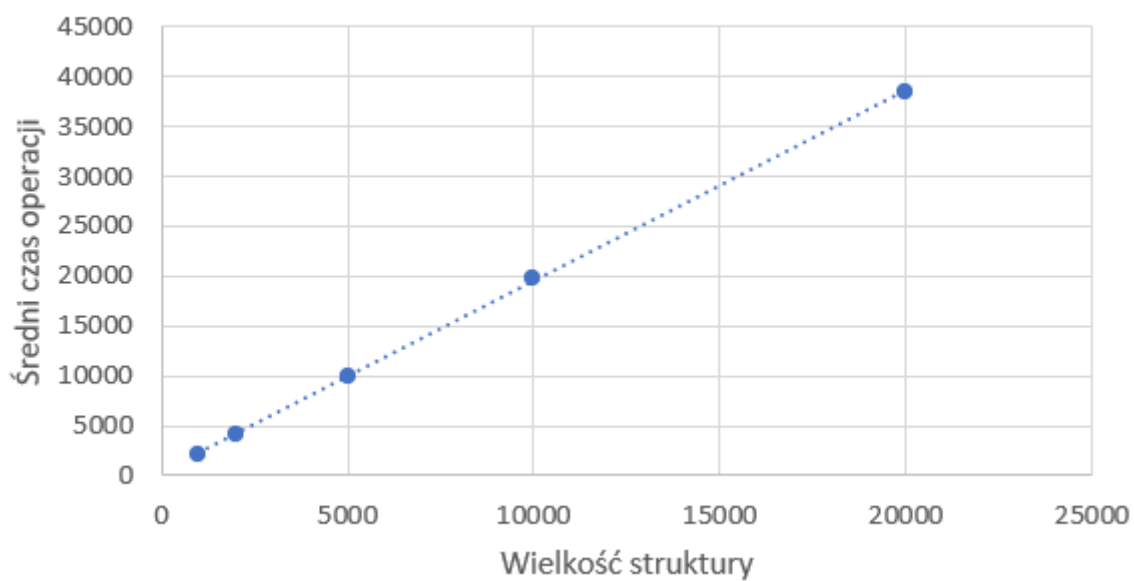
Dodawanie na wybrana pozycje



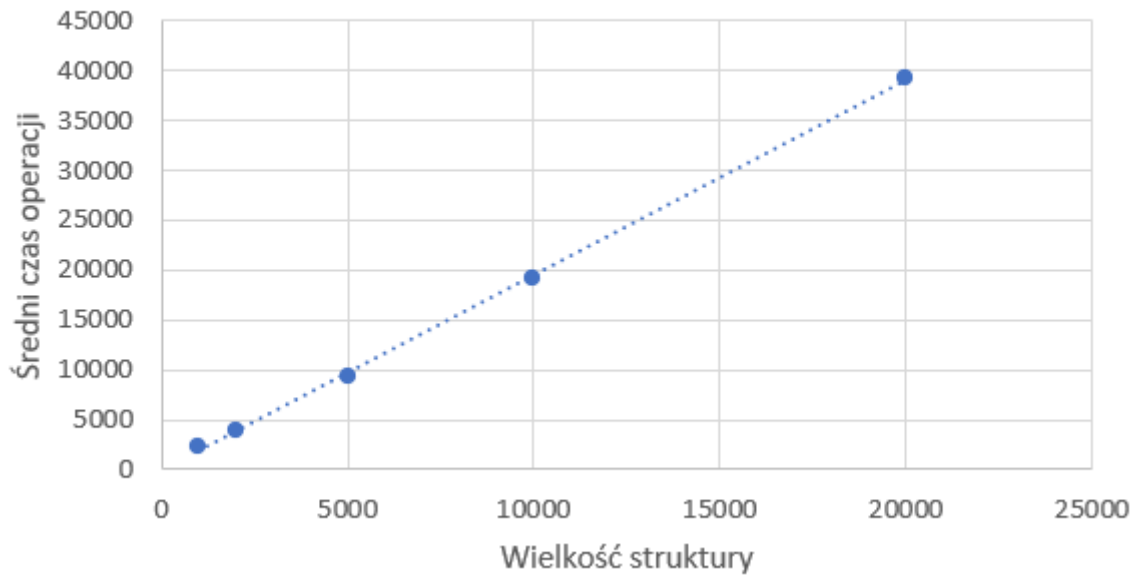
Usuwanie pierwszego elementu



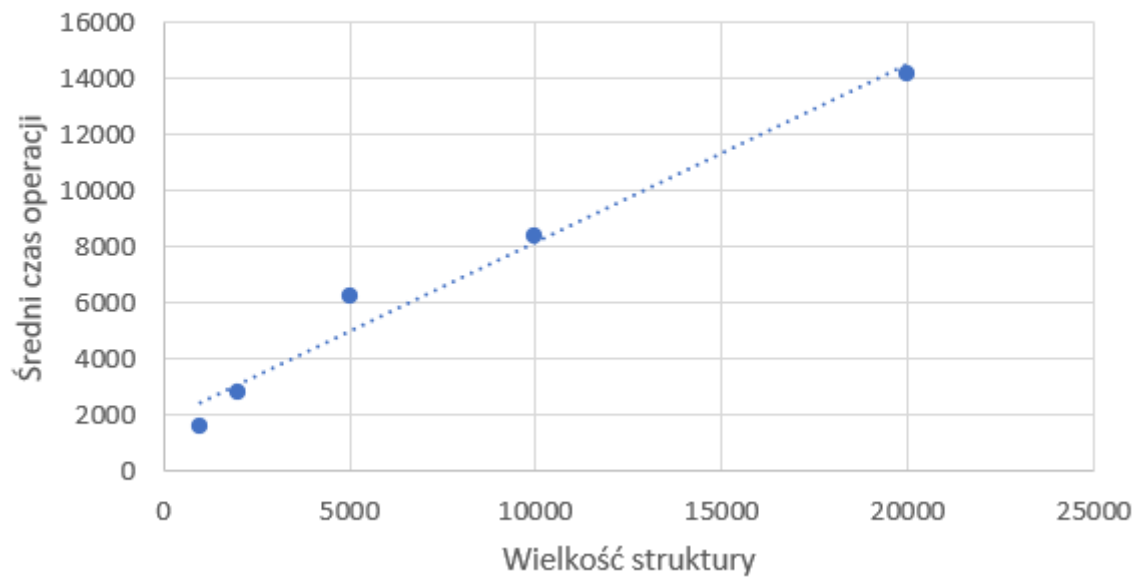
Usuwanie ostatniego elementu



Usuwanie wybranego elementu



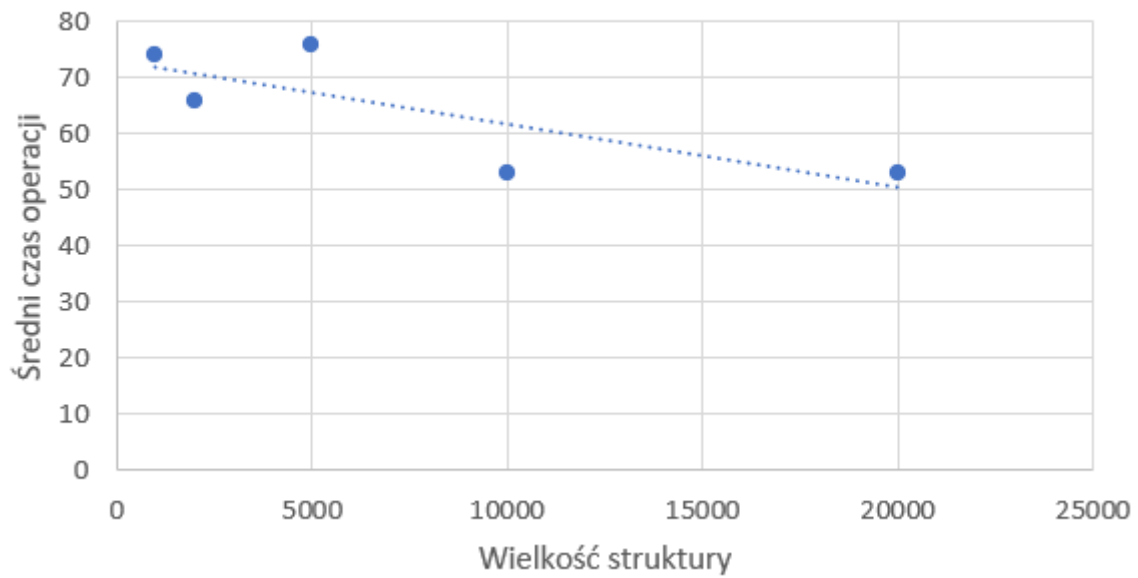
Wyszukiwanie elementu



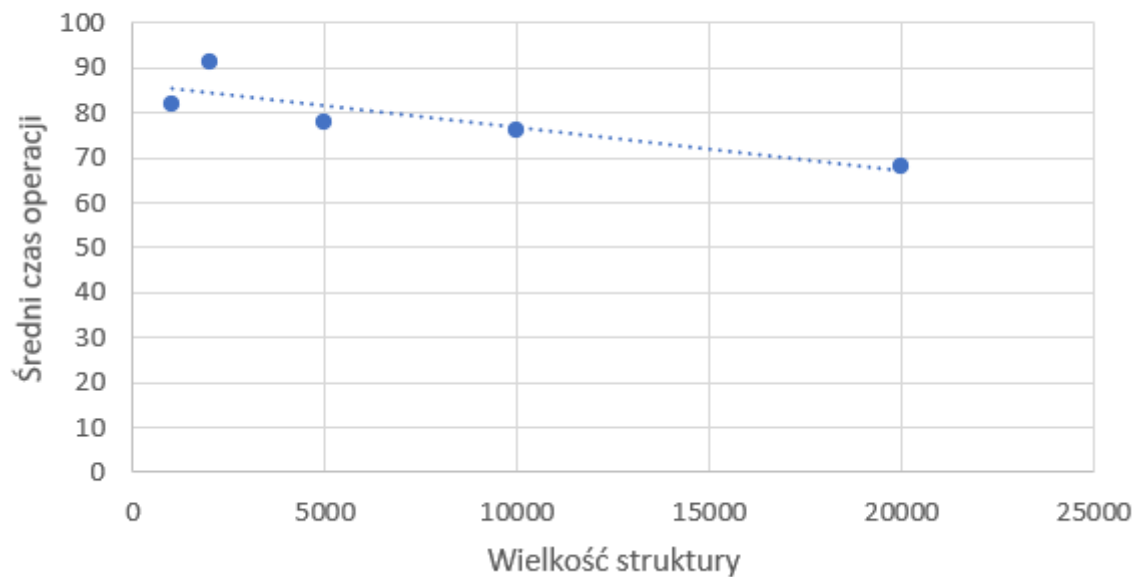
Wnioski: W ostatnim przypadku wykres nie jest idealnie liniowy. Może to wynikać z faktu, iż w tablicy dane nie są w żaden sposób uporządkowane, więc czasy ich wyszukiwani mogą być albo stosunkowo bardzo szybkie albo stosunkowo bardzo długie, w zależności od tego jak rozłożą się dane w strukturze.

4.2. Lista dwukierunkowa

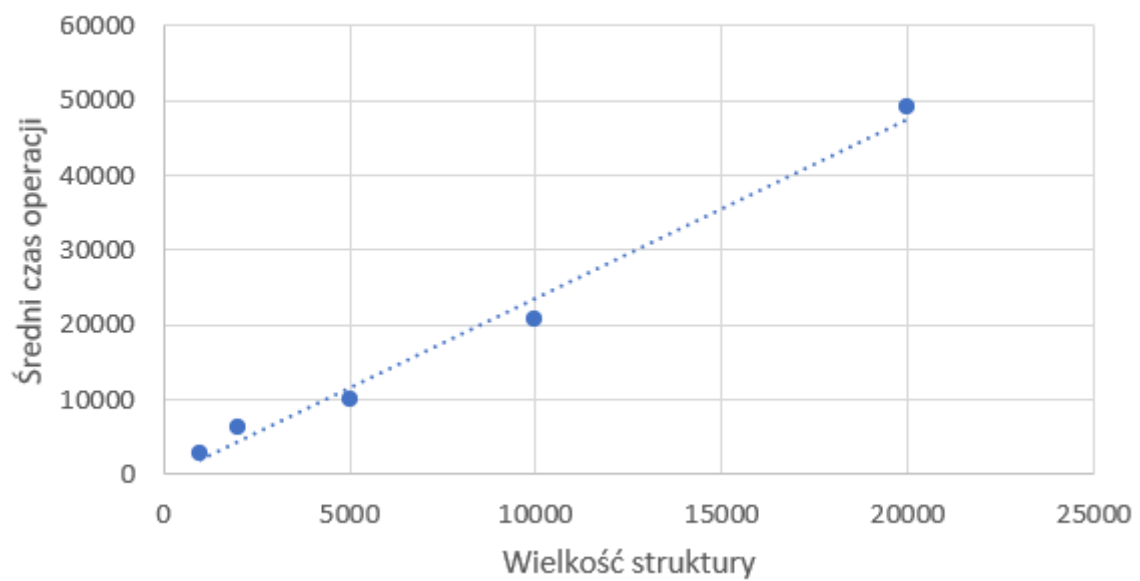
Dodawanie na początek



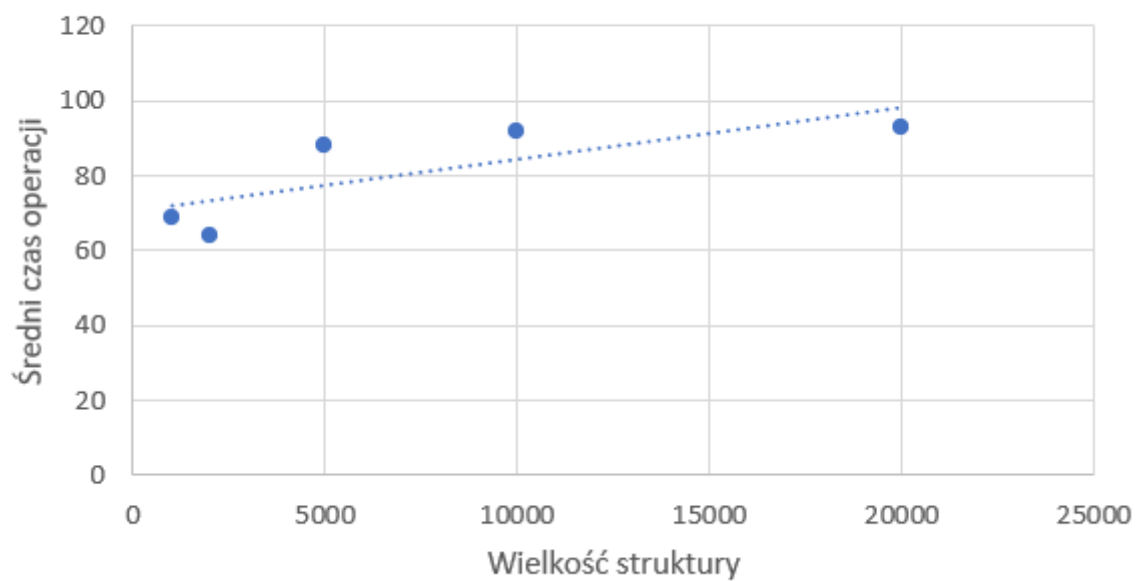
Dodawanie na koniec



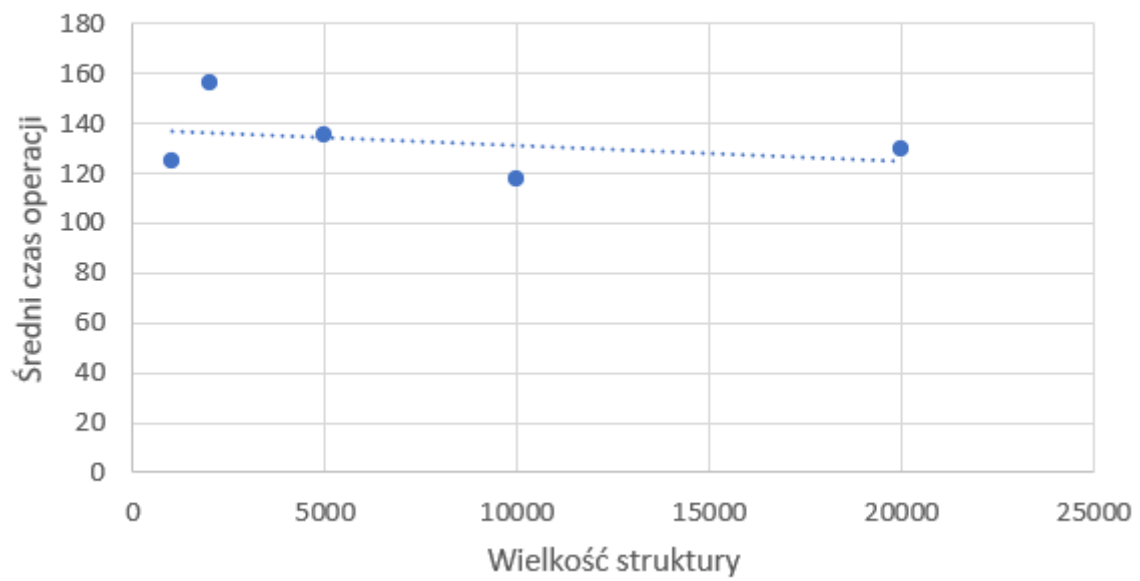
Dodawanie na wybrana pozycje



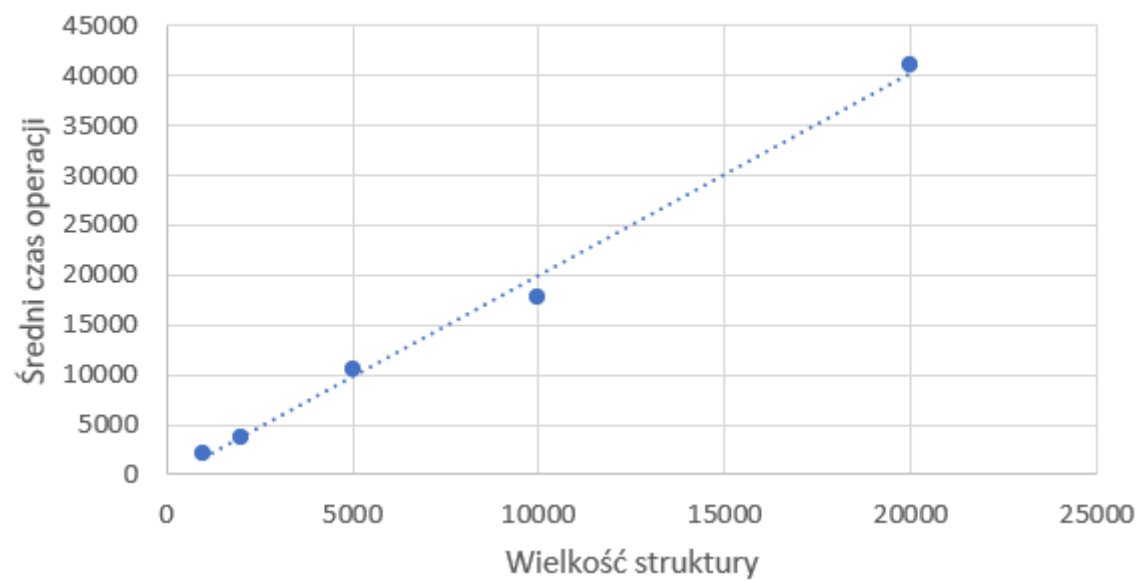
Usuwanie pierwszego elementu



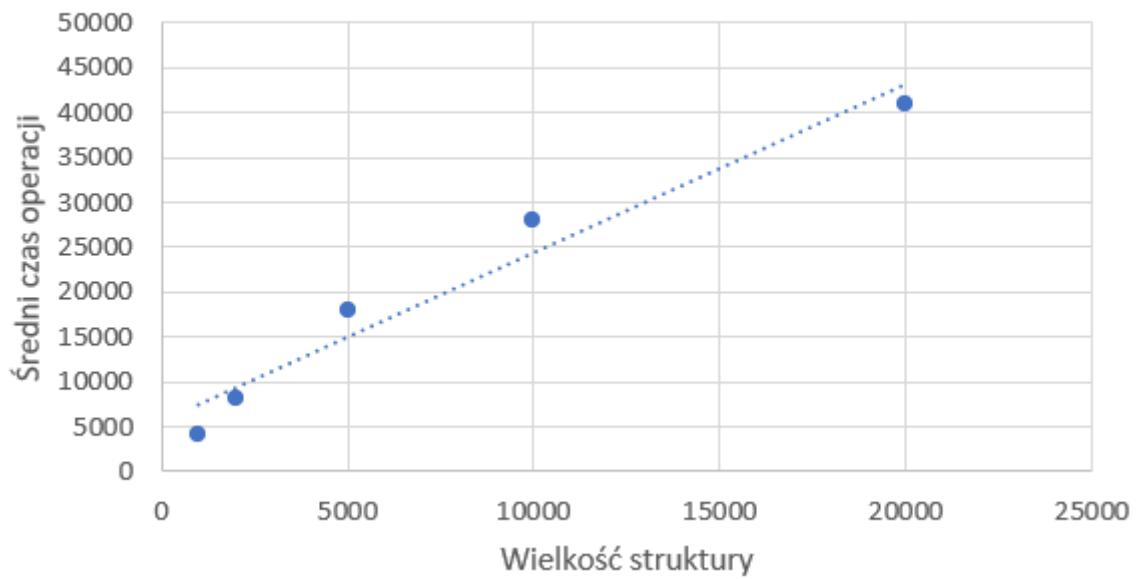
Usuwanie ostatniego elementu



Usuwanie wybranego elementu



Wyszukiwanie elementu

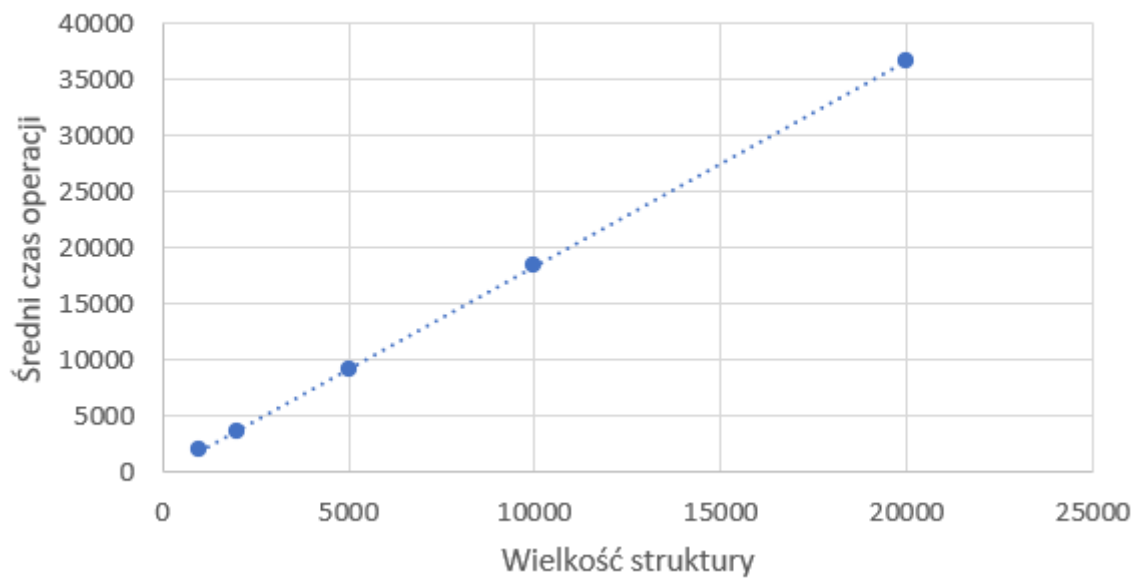


Wnioski: odchylenia od stałej wartości złożoności w części wykresów mogą wynikać z ograniczeń sprzętowych.

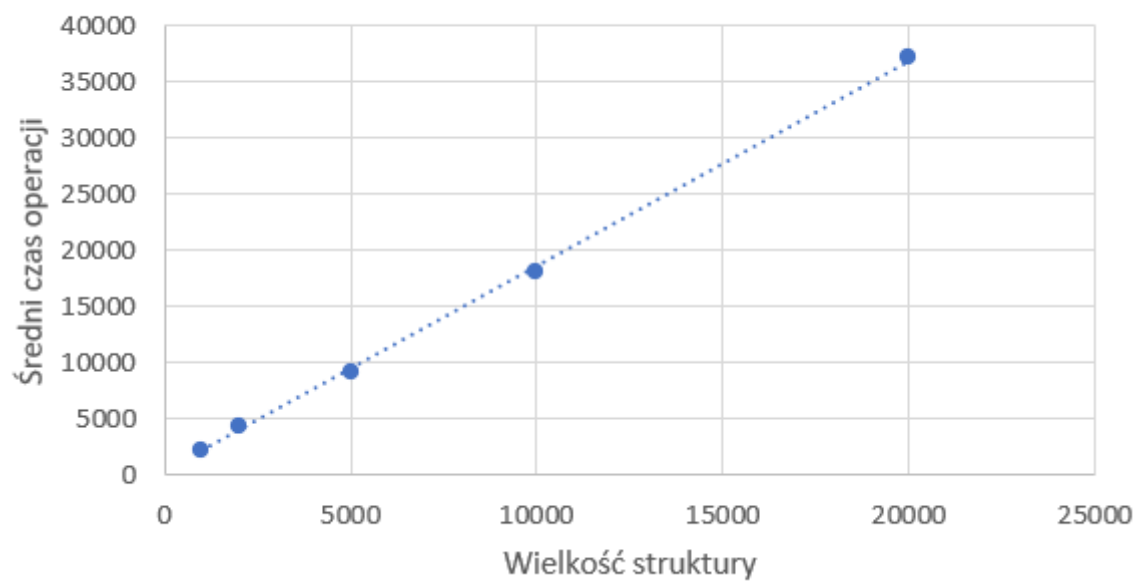
4.3. Kopiec binarny

Operacja/Wielkość struktury	1000	2000	5000	10000	20000
DODAWANIE ELEMENTU	2083	3699	9215	18409	36 722
USUWANIE ELEMENTU	2253	4323	9275	18152	37128
WYSZUKIWANIE ELEMENTU	1682	2732	6233	11170	21099

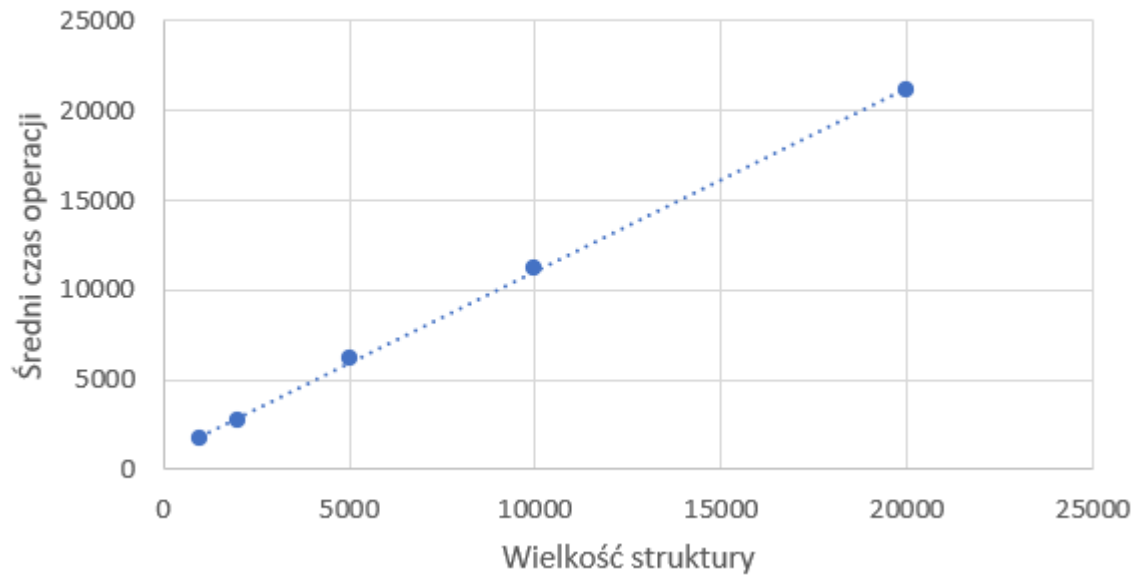
Dodawanie elementu



Usuwanie elementu



Wyszukiwanie elementu



Wnioski: Jak widać na załączonych wyżej obrazkach kształty wykresów dodawania i odejmowania sugerują raczej złożoność obliczeniową $O(n)$, a nie $O(\log n)$. Może to wynikać ze sposobu implementacji struktury. W tym przypadku była to tablica alokowana dynamicznie, która wymagała relokacji w operacjach dodawania i odejmowania elementów.