

# Sprawozdanie SDiZO

*Michał Malewicz 259509*

Zadane zostało stworzenie czterech struktur danych – listy dwukierunkowej, tablicy dynamicznej, kopca binarnego oraz drzewa czerwono-czarnego a następnie przetestowanie złożoności czasowej poszczególnych metod tych kontenerów. Drzewo czerwono-czarne nie zostało zawarte w badaniu.

## Badane metody oraz ich kontenery wraz z opisami złożoności czasowej:

### Lista dwukierunkowa

- a) Dodawanie elementu na początek listy (add front)
- b) Dodawanie elementu na koniec listy (add rear)
- c) Dodawanie elementu w dowolnym miejscu listy (add on position)
- d) Usuwanie elementu z początku listy (delete front)
- e) Usuwanie elementu z końca listy (delete rear)
- f) Usuwanie elementu z dowolnego miejsca listy (delete on position)
- g) Znajdywanie pierwszego elementu równemu podanemu (find first)

Wszelkie operacje wymagające dostępu do zarówno pierwszego jak i ostatniego elementu listy dwukierunkowej będą miały złożoność  $O(1)$ , ze względu na stały dostęp listy do jej początku jak i końca. Pozostałe operacje takie jak dodawanie i usuwanie elementów ze środka listy, lub wyszukiwanie elementu, będą miały złożoność pesymistyczną  $O(n)$  wynikającą z braku uporządkowania listy.

### Tablica dynamiczna

- a) Dodawanie elementu na początek listy (add front)
- b) Dodawanie elementu na koniec listy (add rear)
- c) Dodawanie elementu w dowolnym miejscu listy (add pos)
- d) Usuwanie elementu z początku listy (delete front)
- e) Usuwanie elementu z końca listy (delete rear)
- f) Usuwanie elementu z dowolnego miejsca listy (delete on position)
- g) Znajdywanie pierwszego elementu równemu podanemu (find first)

Każda operacja dodawania oraz odejmowania na tablicy dynamicznej będzie miała pesymistyczną złożoność wynoszącą  $O(n)$  spowodowaną potrzebą kopiowania każdego elementu tablicy. Operacja wyszukiwania elementu także będzie miała złożoność  $O(n)$  ze względu na brak uporządkowania tablicy.

## Kopiec Binarny

- a) Dodawanie elementu (add)
- b) Usuwanie elementu (delete)
- c) Znajdywanie pierwszego elementu równemu podanemu (find first)

Operacja dodawania oraz odejmowania powinna mieć złożoność  $O(\lg(n))$  a wyszukiwania  $O(n)$ , ale ze względu na oparcie struktury na tablicy dynamicznej średnia złożoność obu operacji wynosić będzie  $O(n)$ .

## Przebieg badania:

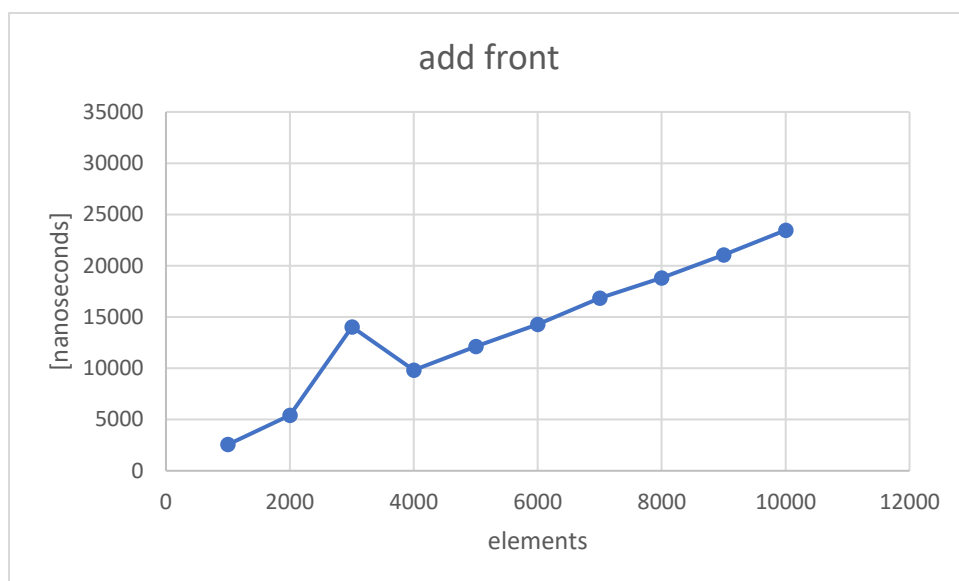
Analiza złożoności czasowej poszczególnych struktur polegać będzie na zestawieniu ze sobą średnich czasów wykonywania metod dla tysiąca powtórzeń dla różnych rozmiarów tablic – od 1000 do 10000 ze skokiem rozmiaru co 1000.

Do kontenerów danych losowane będą liczby z przedziału  $\langle -100, 100 \rangle$ , a pozycja na której dodawane bądź usuwane będą dane, losowana będzie przy każdym powtórzeniu operacji dodawania lub usuwania. Czas wszystkich operacji będzie sumowany a następnie uśredniany dla danej funkcji dla danej wielkości struktury. Będzie on następnie porównany z analogicznymi metodami dla pozostałych kontenerów danych.

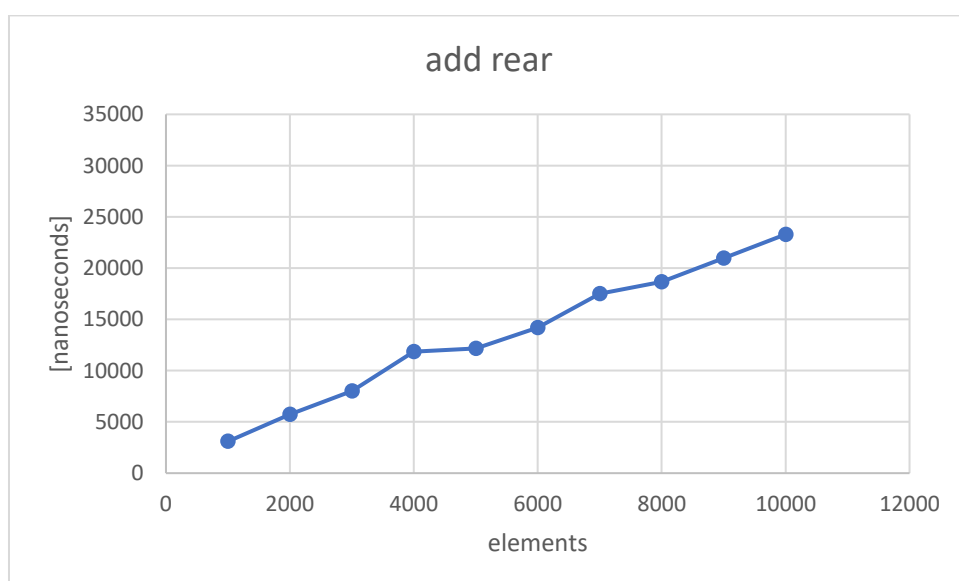
## Uzyskane dane wraz z wykresami reprezentującymi zmiany czasu względem wielkości tablic (czas podawany w nanosekundach):

Tablica dynamiczna

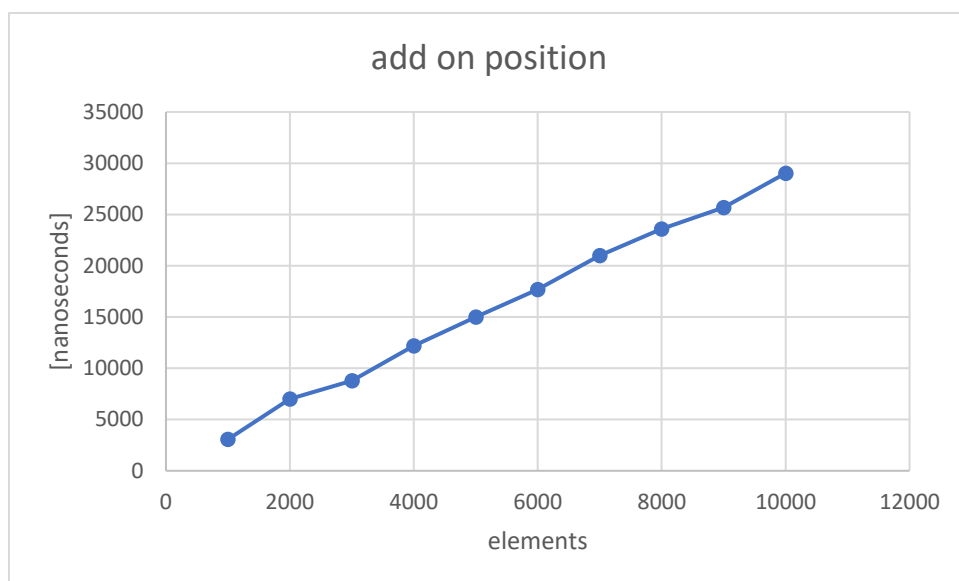
| Liczba obiektów | add front | add rear | add on position | delete front | delete rear | delete pos | find first |
|-----------------|-----------|----------|-----------------|--------------|-------------|------------|------------|
| 1000            | 2558      | 3095,8   | 3065,7          | 2682,1       | 2781        | 3121,4     | 146,3      |
| 2000            | 5399,3    | 5732,2   | 7017            | 5161,1       | 5307,2      | 5947,8     | 129,6      |
| 3000            | 14038,5   | 8003,7   | 8777,1          | 7680,7       | 7660,7      | 8888       | 879,2      |
| 4000            | 9820,5    | 11862,8  | 12180,5         | 10200,8      | 10205,8     | 11822,8    | 853,2      |
| 5000            | 12136,7   | 12160,5  | 15010,8         | 12839,3      | 12677,6     | 14798,6    | 948,2      |
| 6000            | 14279,9   | 14208,8  | 17671,9         | 15110,7      | 15227,1     | 17822      | 931,2      |
| 7000            | 16846,1   | 17501,7  | 21005,6         | 17738,5      | 17405,7     | 20174,6    | 1005,3     |
| 8000            | 18818,3   | 18650,1  | 23591,7         | 19898,6      | 19809,6     | 23396,3    | 1061,8     |
| 9000            | 21066,7   | 20981,6  | 25664,5         | 22268,5      | 22334,3     | 25773,4    | 1092,1     |
| 10000           | 23465,7   | 23291,3  | 29021,6         | 24672,1      | 24751,2     | 29688,1    | 1098,2     |



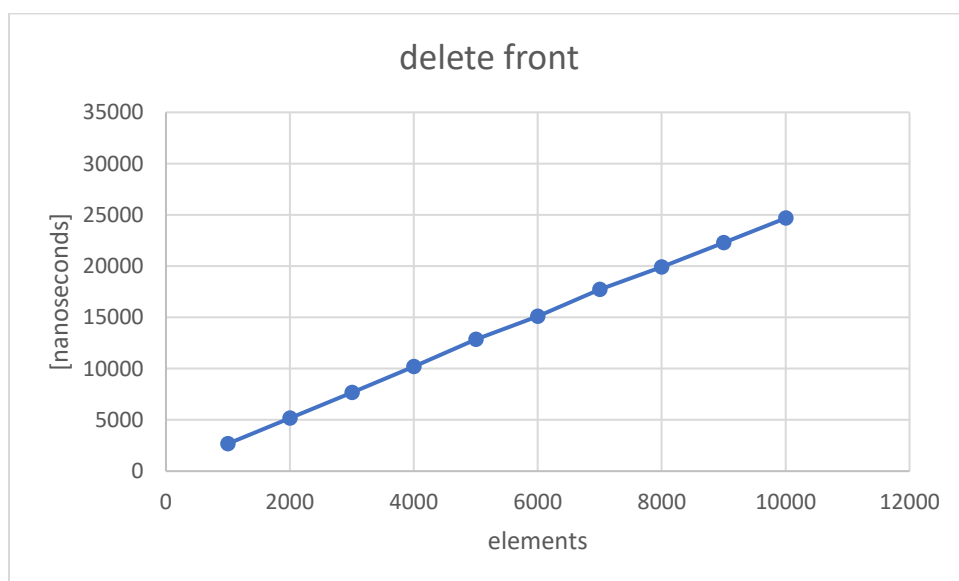
PRZEWIDYWANA ZŁOŻONOŚĆ CZASOWA:  $O(N)$



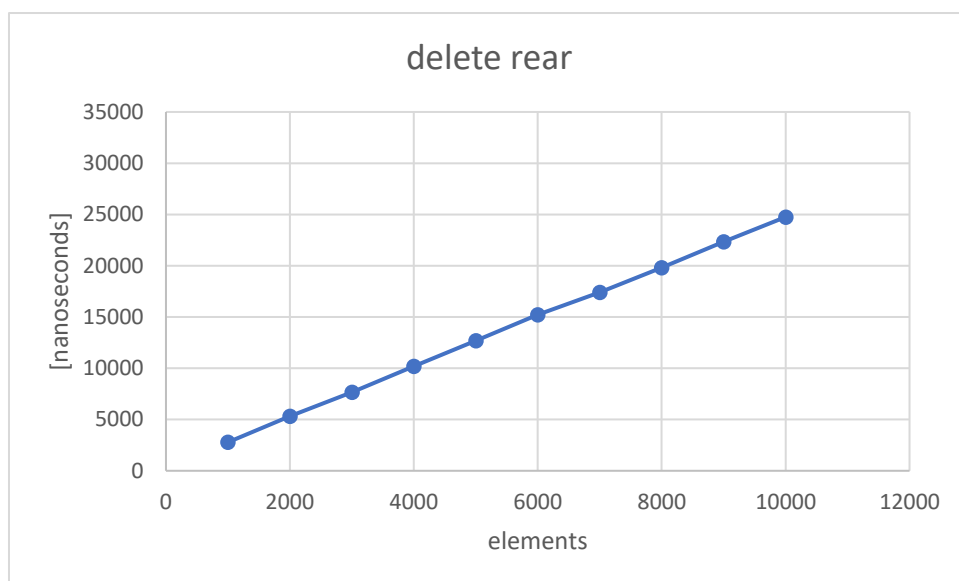
PRZEWIDYWANA ZŁOŻONOŚĆ CZASOWA:  $O(N)$



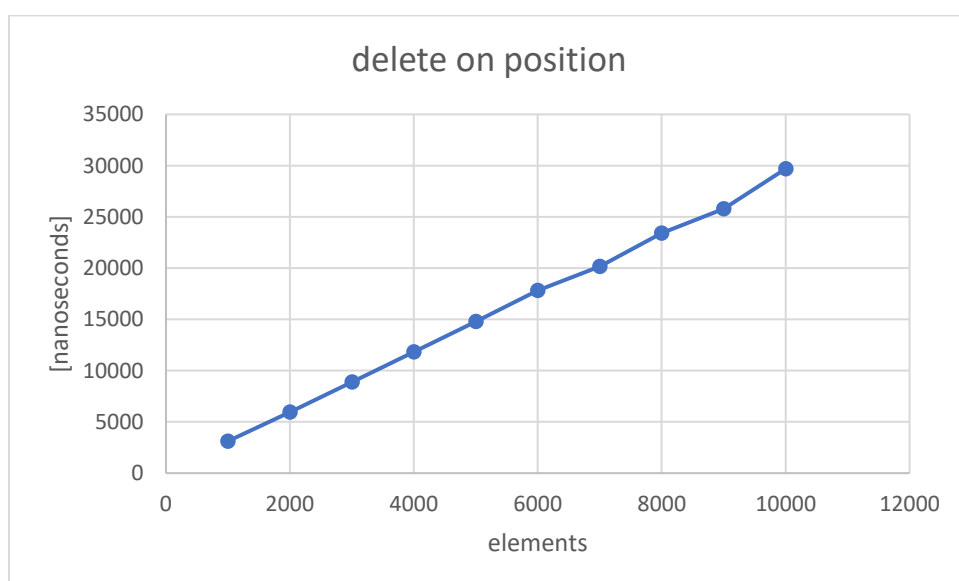
PRZEWIDYWANA ZŁOŻONOŚĆ CZASOWA:  $O(N)$



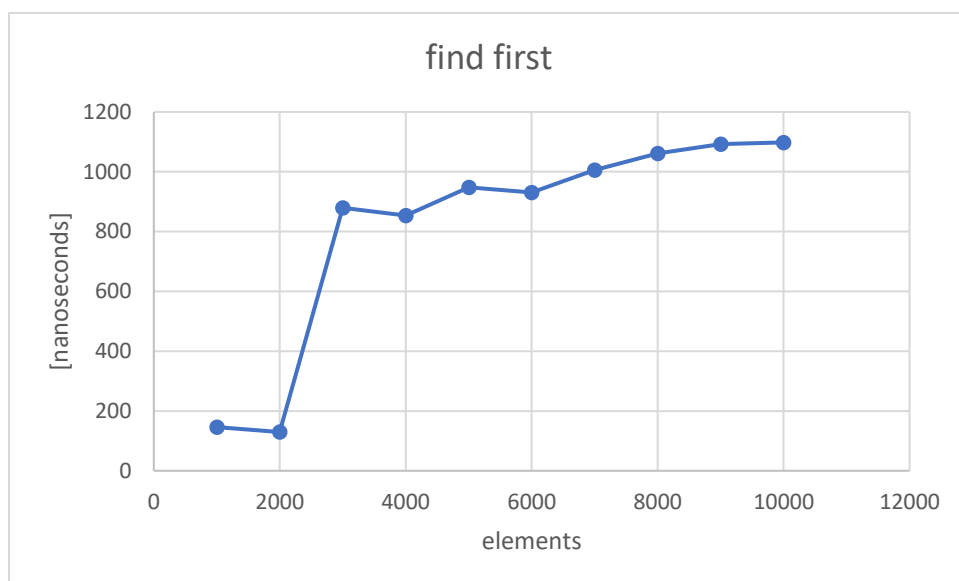
PRZEWIDYWANA ZŁOŻONOŚĆ CZASOWA:  $O(N)$



PRZEWIDYWANA ZŁOŻONOŚĆ CZASOWA:  $O(N)$



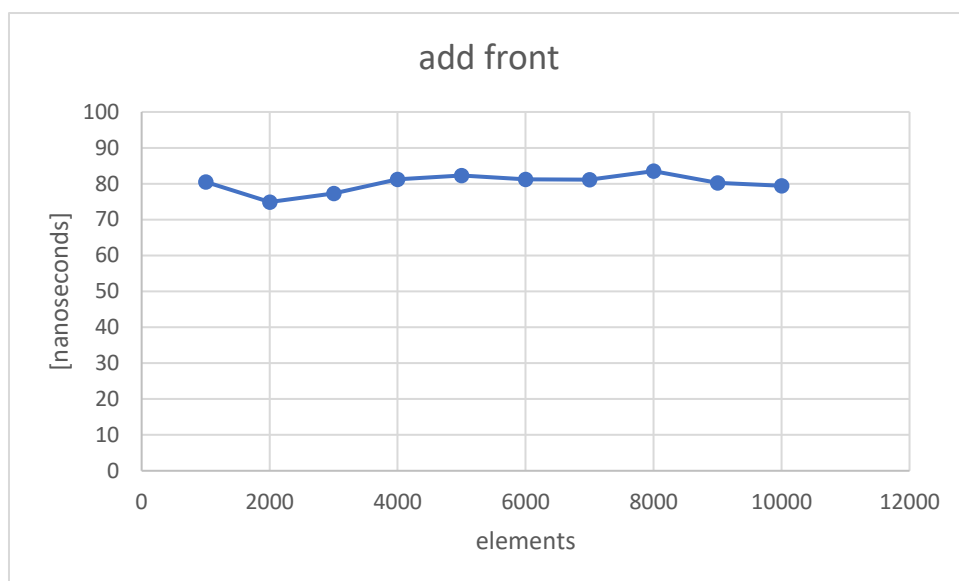
PRZEWIDYWANA ZŁOŻONOŚĆ CZASOWA:  $O(N)$



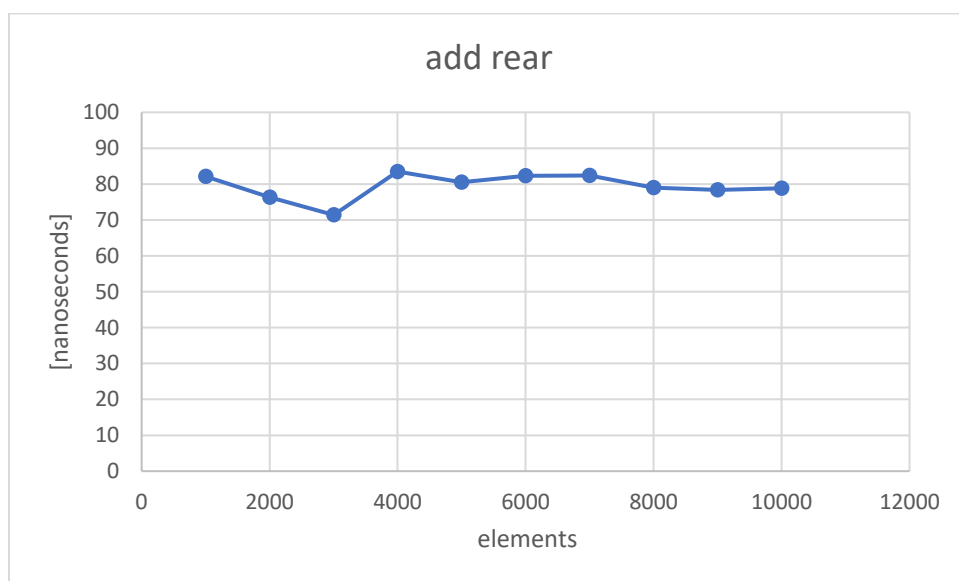
PRZEWIDYWANA ZŁOŻONOŚĆ CZASOWA:  $O(N)$

Lista dwukierunkowa

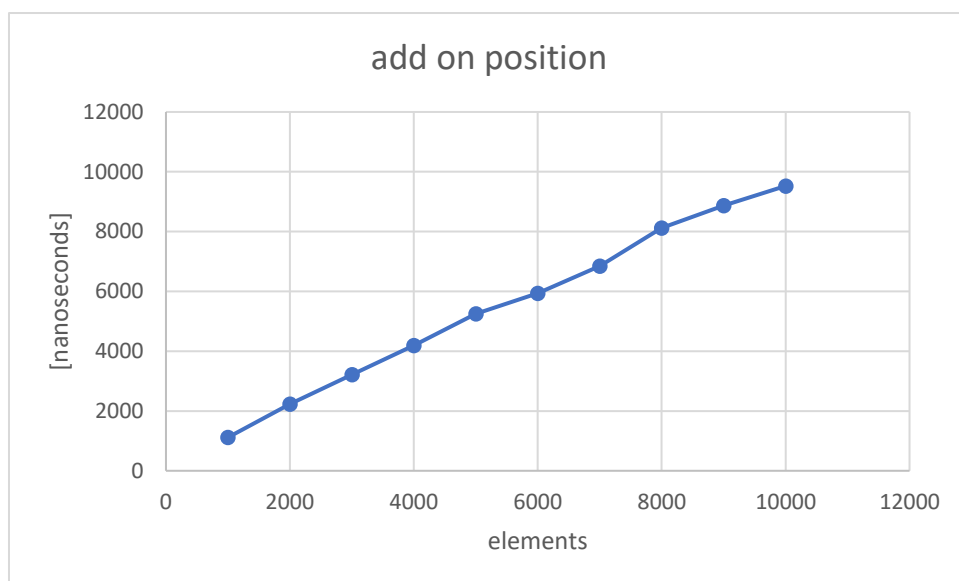
| Liczba obiektów | add front | add rear | add on position | delete front | delete rear | delete pos | find first |
|-----------------|-----------|----------|-----------------|--------------|-------------|------------|------------|
| 1000            | 80,5      | 82,1     | 1115,3          | 70,8         | 65,7        | 1143       | 368,4      |
| 2000            | 74,9      | 76,3     | 2227,1          | 73,9         | 61,4        | 2095,2     | 257,1      |
| 3000            | 77,3      | 71,4     | 3218,3          | 75,1         | 69,9        | 3266,7     | 2934,5     |
| 4000            | 81,2      | 83,5     | 4186,1          | 69,6         | 60,7        | 4033,1     | 3004,3     |
| 5000            | 82,3      | 80,5     | 5254,5          | 70,1         | 62,2        | 5050,1     | 2934,9     |
| 6000            | 81,2      | 82,3     | 5933,6          | 70,7         | 61          | 5716       | 2929,2     |
| 7000            | 81,1      | 82,4     | 6844,9          | 79,3         | 63,8        | 7116,7     | 2970       |
| 8000            | 83,5      | 79       | 8123            | 68,9         | 62,5        | 7702       | 2959,4     |
| 9000            | 80,2      | 78,4     | 8870,8          | 73,1         | 62,5        | 8609,2     | 2956,1     |
| 10000           | 79,4      | 78,8     | 9524,3          | 73,7         | 64,8        | 10190,7    | 2944,8     |



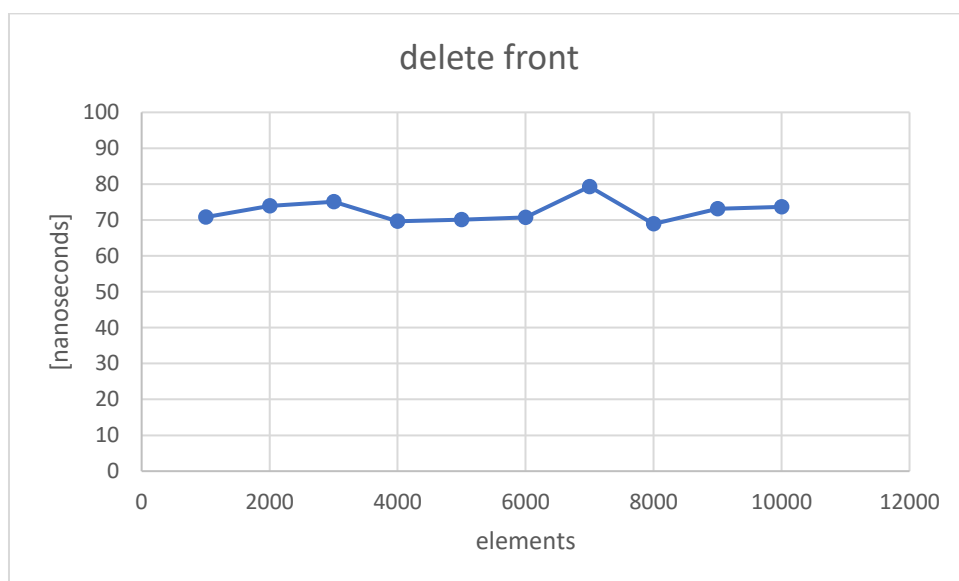
PRZEWIDYWANA ZŁOŻONOŚĆ CZASOWA:  $O(1)$



PRZEWIDYWANA ZŁOŻONOŚĆ CZASOWA:  $O(1)$

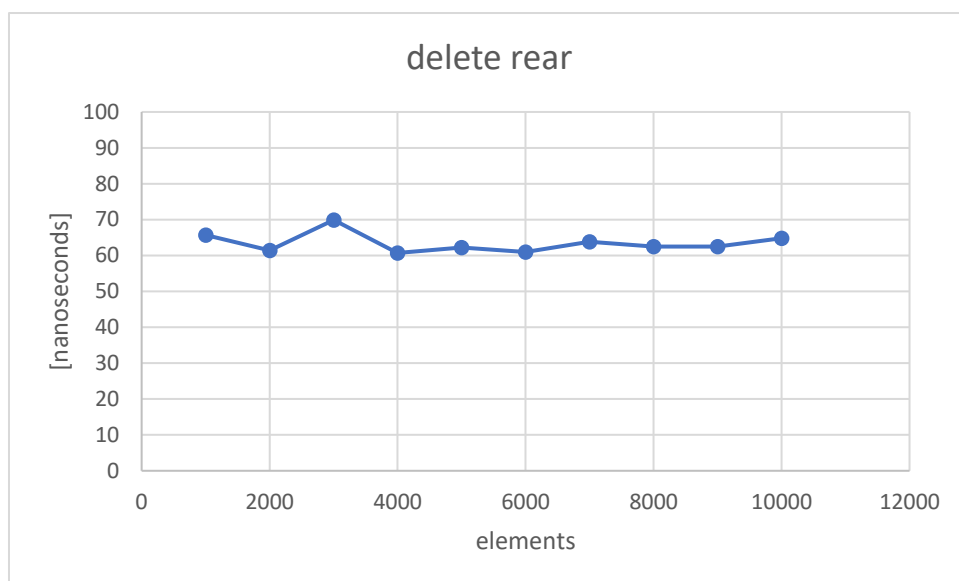


PRZEWIDYWANA ZŁOŻONOŚĆ CZASOWA:  $O(N)$

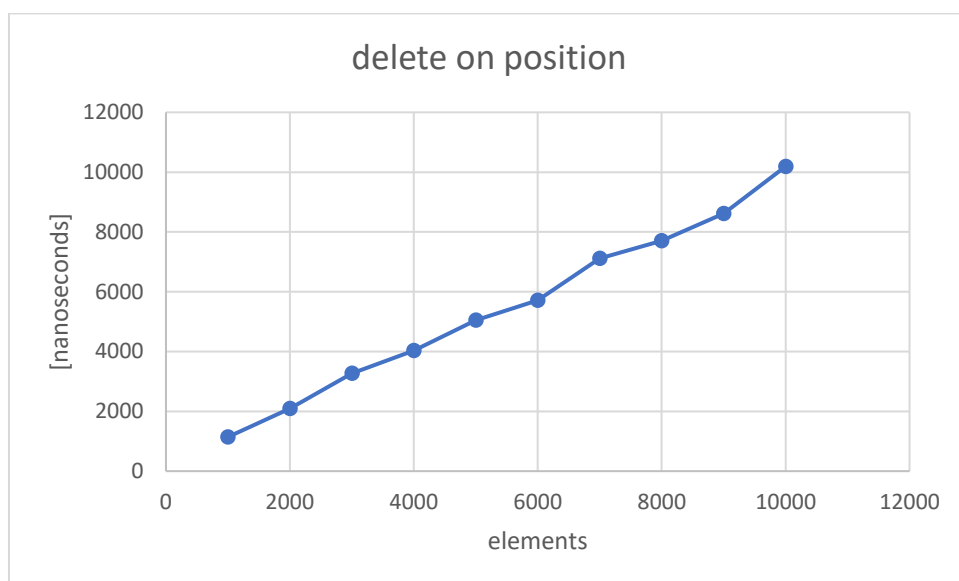


PRZEWIDYWANA ZŁOŻONOŚĆ CZASOWA:  $O(1)$

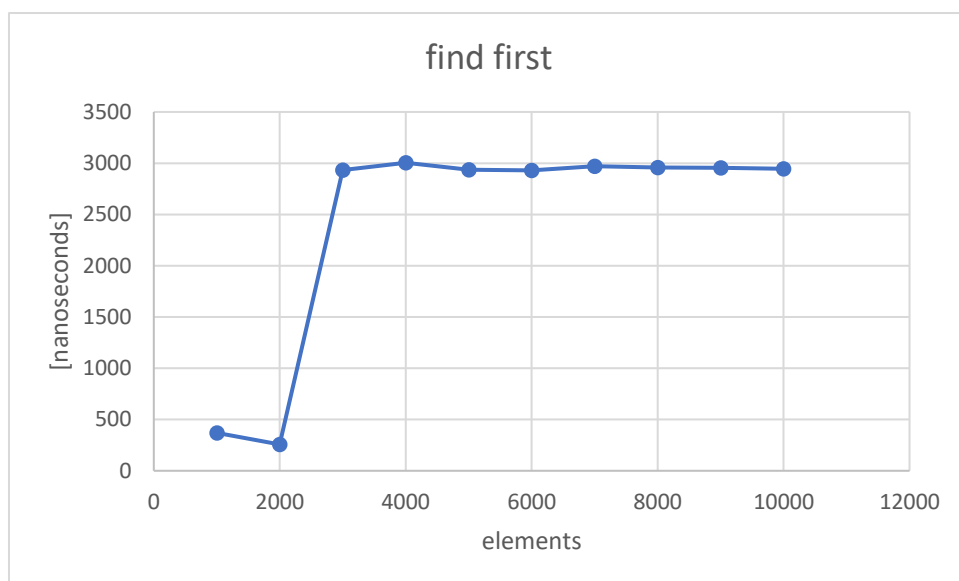




PRZEWIDYWANA ZŁOŻONOŚĆ CZASOWA:  $O(1)$



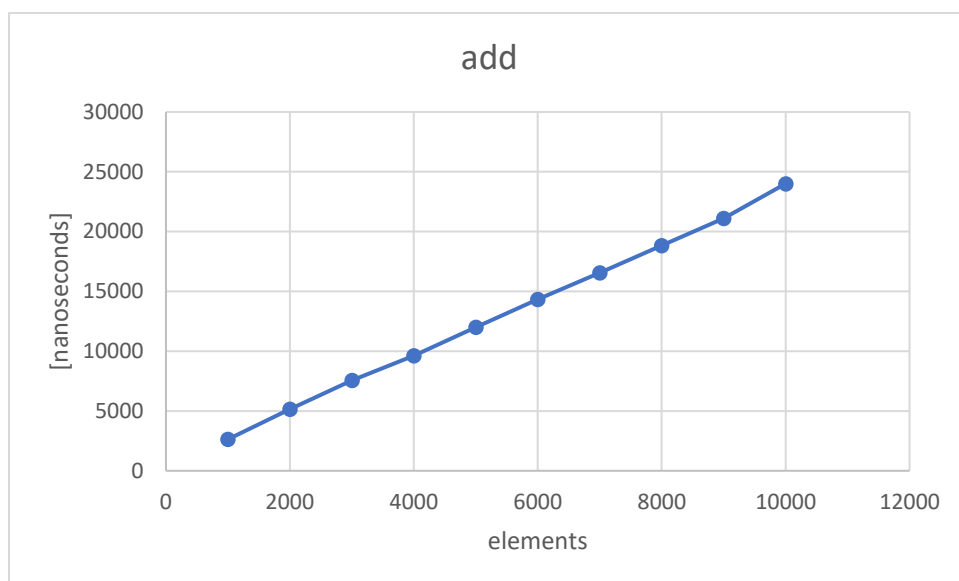
PRZEWIDYWANA ZŁOŻONOŚĆ CZASOWA:  $O(N)$



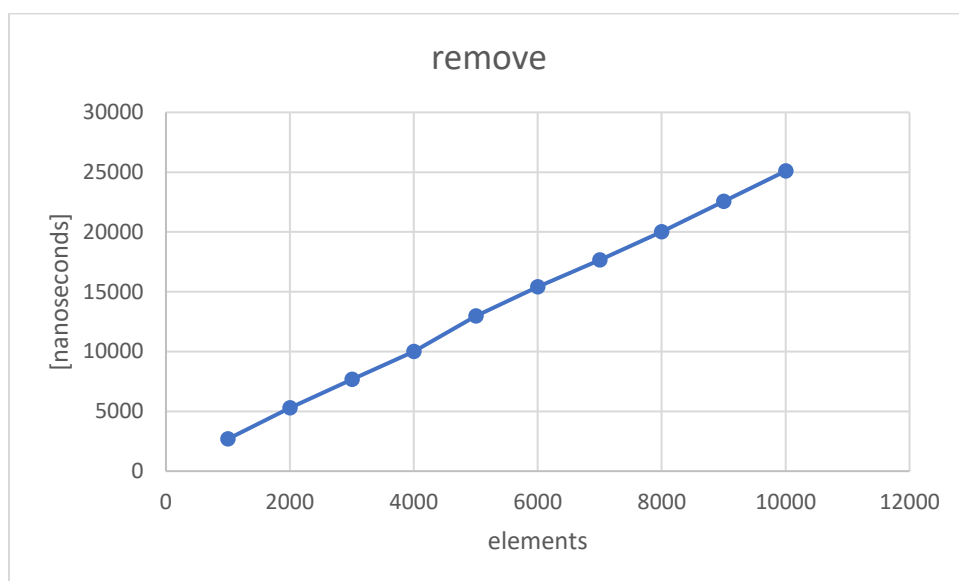
PRZEWIDYWANA ZŁOŻONOŚĆ CZASOWA:  $O(N)$

Kopiec binarny

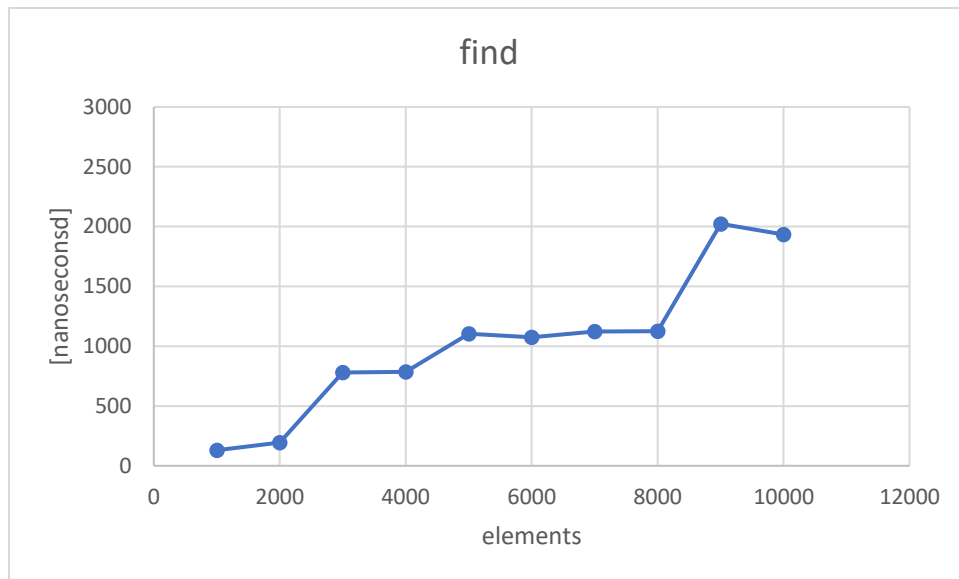
| Liczba obiektów | add     | remove  | find   |
|-----------------|---------|---------|--------|
| 1000            | 2630,9  | 2692,8  | 130,2  |
| 2000            | 5141,6  | 5287,9  | 194,3  |
| 3000            | 7556    | 7660,3  | 780,2  |
| 4000            | 9617,2  | 9987,3  | 786,4  |
| 5000            | 12003   | 12976,3 | 1105   |
| 6000            | 14319,3 | 15416,7 | 1073,3 |
| 7000            | 16552,5 | 17663,8 | 1122,5 |
| 8000            | 18824,2 | 20015,1 | 1125,7 |
| 9000            | 21111,7 | 22549,4 | 2022,8 |
| 10000           | 24001,8 | 25092,8 | 1934,5 |



PRZEWIDYWANA ZŁOŻONOŚĆ CZASOWA:  $O(N)$



PRZEWIDYWANA ZŁOŻONOŚĆ CZASOWA:  $O(N)$



**PRZEWIDYWANA ZŁOŻONOŚĆ CZASOWA:  $O(N)$**

## Wnioski:

Wyniki badań zmiany złożoności czasowej w większości pokrywają się z zakładanymi dla danych struktur. Wszelkie różnice wynikać mogą w szczególności z ograniczeń sprzętowych (zakłóceń pracy komputera).

Najwydajniejszą strukturą do przechowywania oraz wydobywania w szczególności dużych danych okazuje się lista dwukierunkowa. Silną alternatywą mógłby okazać się kopiec binarny, gdyby nie oparcie tego kontenera na konstrukcie tablicy dynamicznej. W tym zestawieniu dużym pozytywem tablicy, mimo jej stosunkowo wolnego czasu pracy okazuje się swoboda pracy na tej strukturze.