

Sprawozdanie PAMSI

Zadanie: Oszacowanie złożoności obliczeniowej wyszukiwania w liście jednokierunkowej

Funkcja przeszukiwania listy:

```
void list_head::find_in_list(int value){
    element* pointer = first;
    while((pointer->next !=0) && (pointer->value != value)){
        pointer = pointer->next;
        //cout<<"1 ";
    }
}
```

przeszukuje listę po wskaźnikach do następnych elementów w poszukiwaniu zadanej value. Przy każdej inicjacji listy szukany element znajduje się na jej końcu.

Spodziewana złożoność obliczeniowa:

$O(n)$ n – ilość elementów

Tabele pomiarowe:

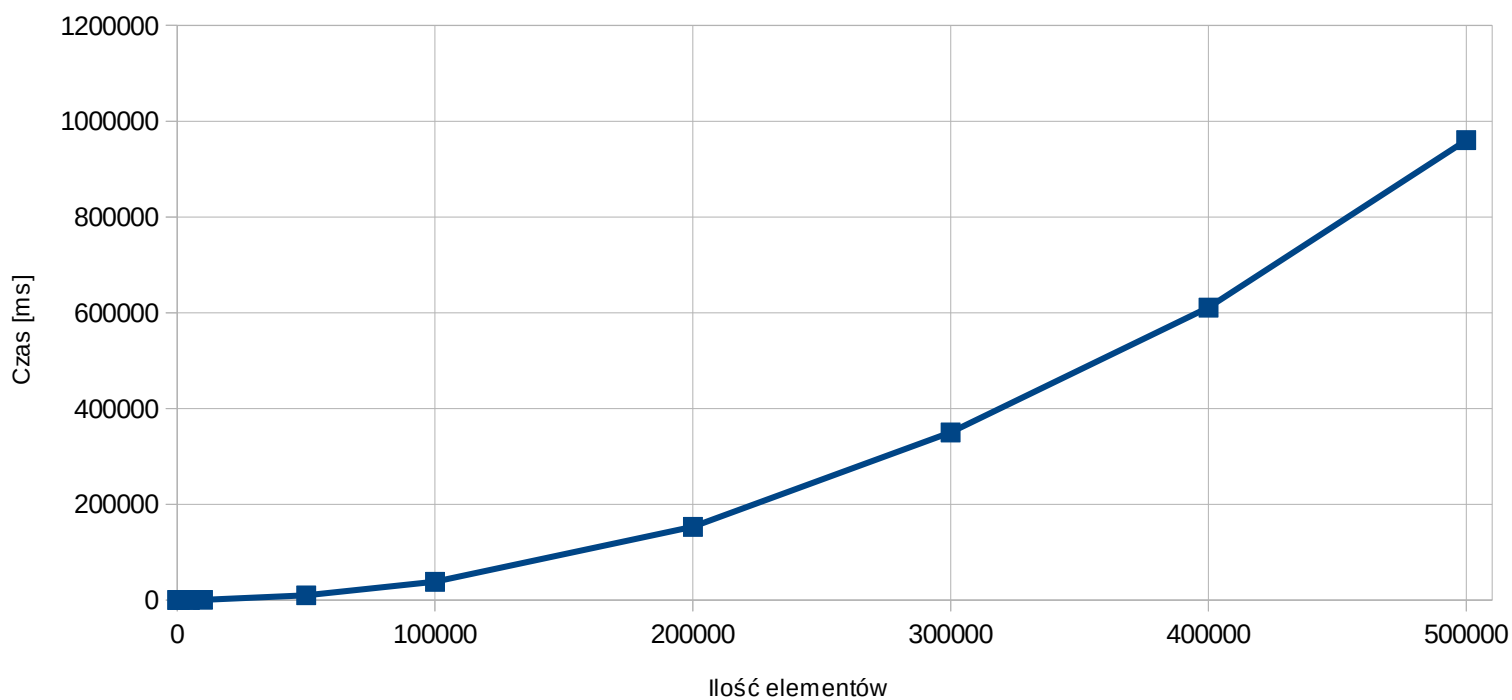
Ilość elementów Listy	Czas [ms]	Ilość elementów Listy	Czas [ms]
10	1	100000	38245
	1		37958
	1		38376
	0		38096
	1		38230
100	1	200000	152224
	1		154563
	1		152874
	1		152350
	1		152324
1000	4	300000	351872
	4		351619
	4		349123
	5		349138
	4		348191
5000	98	400000	611237
	105		610822
	98		611092
	98		612918
	94		611198
10000	381	500000	960342
	386		961056
	380		960976
	374		960315
	382		961242
50000	9571		
	10000		
	10177		
	10147		
	9727		

Tabela 1

Ilość elementów Listy	Średni czas [ms]
10	1
100	1
1000	4
5000	99
10000	381
50000	9924
100000	38181
200000	152867
300000	349989
400000	611453
500000	960786

Tabela 2

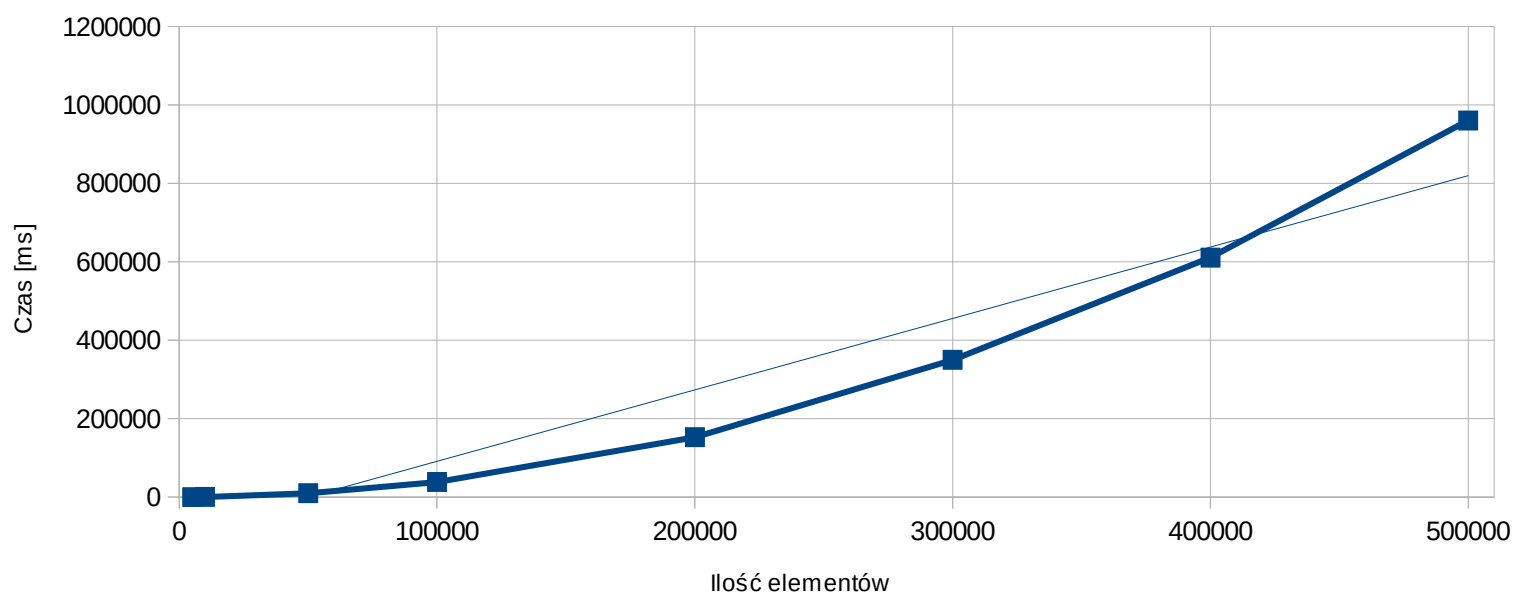
Złożoność obliczeniowa przeszukiwanie listy jednokierunkowej



Wykres 1

Złożoność obliczeniowa

Wykres złożoności obliczeniowej dla najbardziej miarodajnych danych



Wykres 2

Wnioski:

Wyniki dla list o długości: 10, 100, 1000 możemy uznać za zbyt mało wiarygodne, ponieważ dokładność pomiaru czasu przeszukiwania jest w tym przypadku zbyt mała. Na podstawie zebranych danych możemy oszacować złożoność obliczeniową algorytmu przeszukiwania listy bliską teoretycznej złożoności $O(n)$. Bliskość tę widać szczególnie na Wykres 2, gdzie bardziej zostały porównane najbardziej miarodajne dane. Odchylenia pomiędzy teoretyczną złożonością obliczeniową a otrzymaną charakterystyką można wytłumaczyć nieidealnymi warunkami podczas testowania algorytmu.