

SPRAWOZDANIE 3

Przeszukiwanie listy jednokierunkowej

Program zapisuje listę losowymi liczbami z przedziału 0-100, następnie po jej wypełnieniu dokonuje przejścia przez listę w celu znalezienia odpowiedniej (również wybranej losowo) liczby.

Jako, że zwracana jest jedynie liczba elementów pasujących do schematu teoretyczna złożoność obliczeniowa wynosi $O(n)$, gdyż metoda *przeszukaj(int x)* przechodzi po liście złożonej z „n” elementów, porównuje element do wartości szukanej, a następnie wypisuje ilość znalezionych elementów.

1. Czas obliczeń

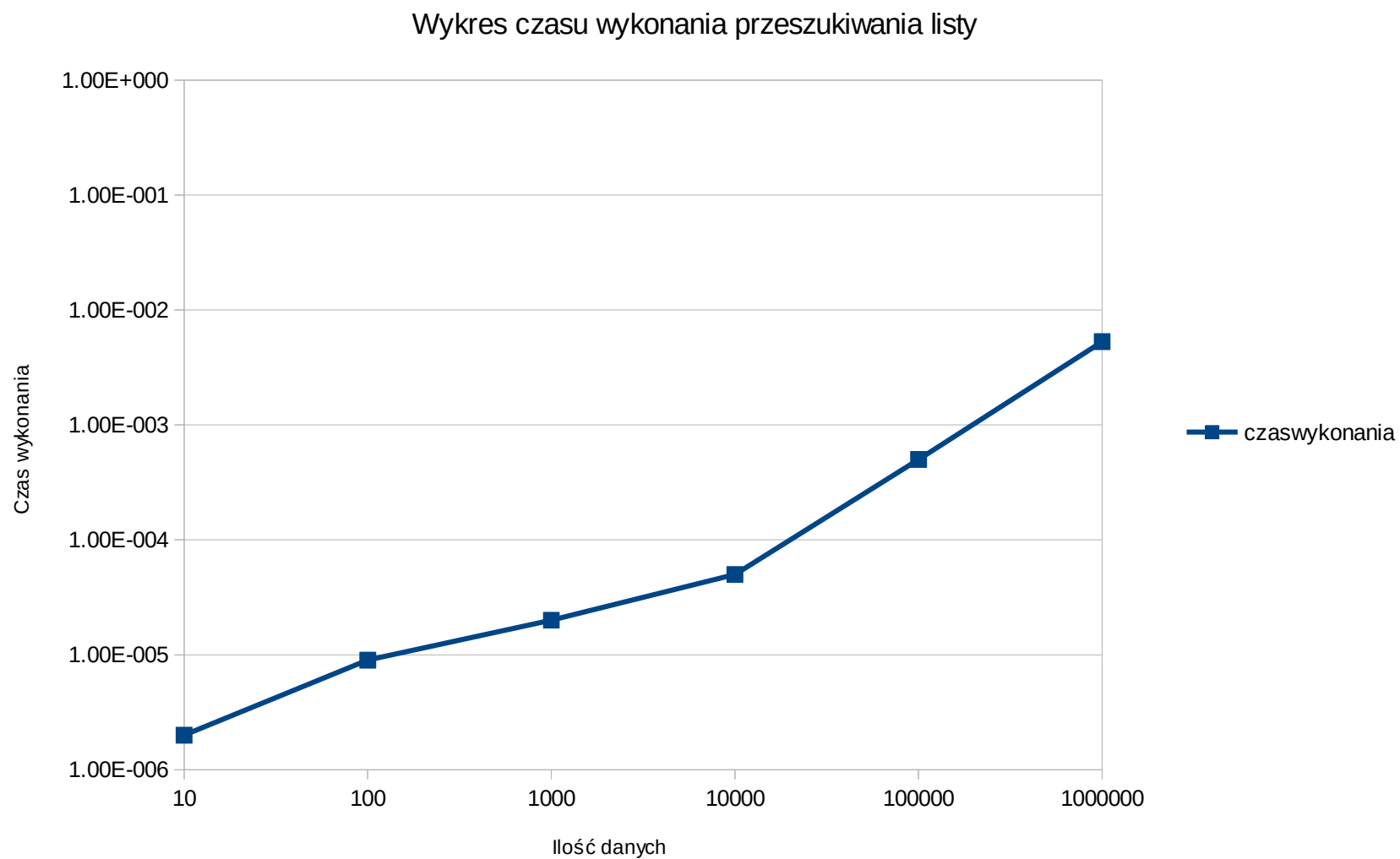
Liczba elementów	Czas obliczeń [s]
10	0,000002
100	0,000009
1000	0,00002
10 000	0,00005
100 000	0,0005
1 000 000	0,0053
1 000 000 000	Nie przeprowadzono pomiaru

Testy wykonywane były na laptopie wyposażonym w procesor Intel i5, a także 6 Gb RAM. Każdy test powtarzany był dziesięciokrotnie, w cel uzyskania jak najdokładniejszego wyniku.

Jako, że lista opiera się na wskaźnikach przejście z jednego elementu do drugiego jest bardzo szybkie, dlatego też czasy rzędu tysięcznej części sekundy pojawiają się dopiero dla listy wielkości miliona elementów, natomiast przy listach od 10 do 10 000 elementowych czasy wykonania były rzędu mikrosekund.

Z powodu ograniczeń czasowych nie udało się przeprowadzić pomiaru dla listy złożonej z miliarda elementów.

2. Wykresy:



Wykres zrealizowano w skali log-log. Nachylenie świadczy o wzroście funkcji

3. Wnioski

Przeszukiwanie tablicy założonego algorytmu ma złożoność $O(n)$, co można łatwo zaobserwować na załączonym wykresie. Drobne zniekształcenie wykresu wiąże się ze stosunkowo małą liczbą powtórzeń obliczeń (10) .