

Projektowanie algorytmów i metody sztucznej inteligencji

Projekt 1

Prowadzący: Dr inż. Andrzej Rusiecki

Wtorek 13:15-15:00

Imię Nazwisko: Mateusz Bartkowiak

Numer indeksu: 249467

1. Wykonanie zadania

Tesowane sortowania :

Przez Scalanie – jest to metoda stosująca metodę dziel i zwyciężaj. Oznacza to, że większy problem rozdzielamy na mniejsze tak aby osiągnąć jeden element bez tak zwanej pary. Algorytm stabilny przy złożoności obliczeniowej $O(n \log n)$.

Quicksort – algorytm, który również korzysta jak poprzednik z metody dziel i zwyciężaj. Różnica jednak jest taka, że wybieramy pivot, który rozdzieli nam zestaw na dwie części. Elementy większe od pivotu przenosimy do części po prawej stronie a mniejsze do lewej. Kluczowe w tym algorytmie jest wybór pivotu dlatego złożoność obliczeniowa jest zależna od przypadku.

Optymistyczny przypadek $O(n \log n)$.

Pesymistyczny przypadek $O(n^2)$.

Introspektywne – metoda hybrydowa, będąca połączeniem sortowania szybkiego i sortowania przez kopcowanie. Złożoność obliczeniowa: $O(n \log n)$.

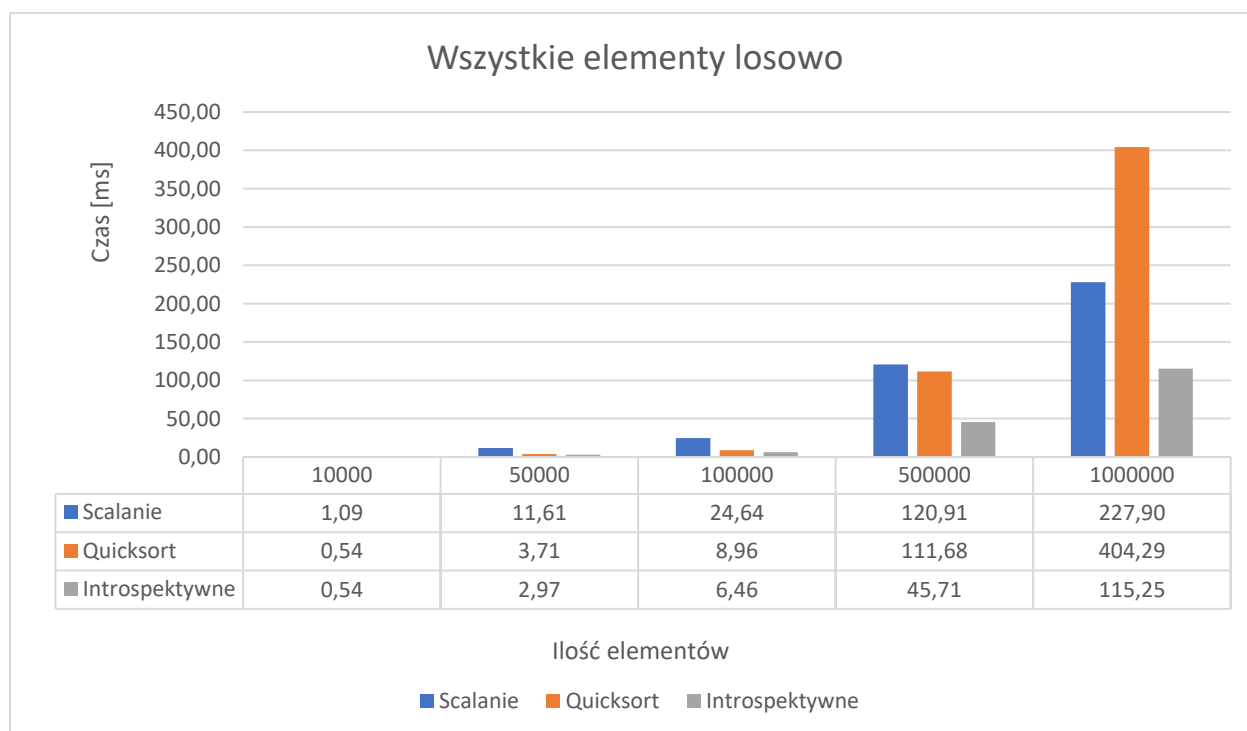
Sortowania zostały wykonane w programie Visual Studio w trybie Release, natomiast czas został zmierzony za pomocą biblioteki time.h. Program generuje liczby, oraz zapisuje czas poszczególnego sortowania do plików tekstowych. Algorytmy przetestowano na różnych wielkościach tablic oraz innych stopniach posortowań początkowych. Pomiary zostały wykonane na laptopie o parametrach:

Pamięć: 8GB RAM

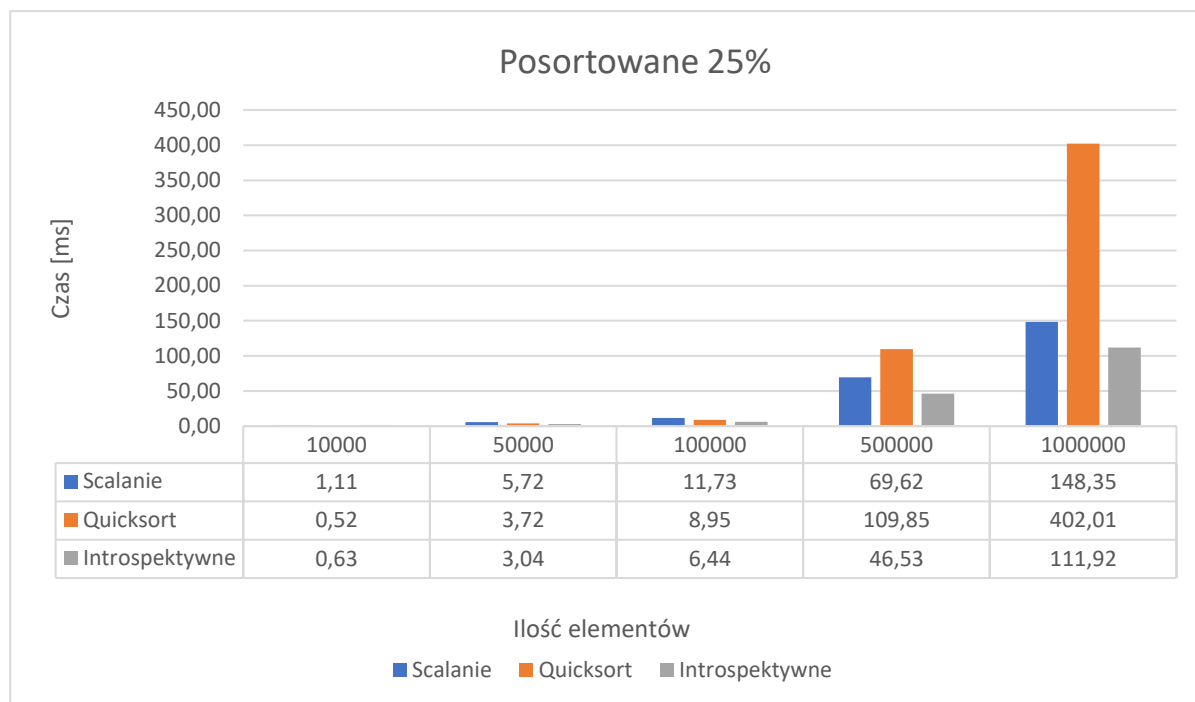
Procesor: Intel Core i5-8250U CPU 1.60Ghz (8 CPUs)

Czas jaki zostanie podany w wynikach jest czasem sortowania jednej tablicy, natomiast jest on policzony z średniej arytmetycznej 100 tablic by uwiarygodnić wynik (program podaje czas wykonania sortowania każdej poszczególniej tablicy).

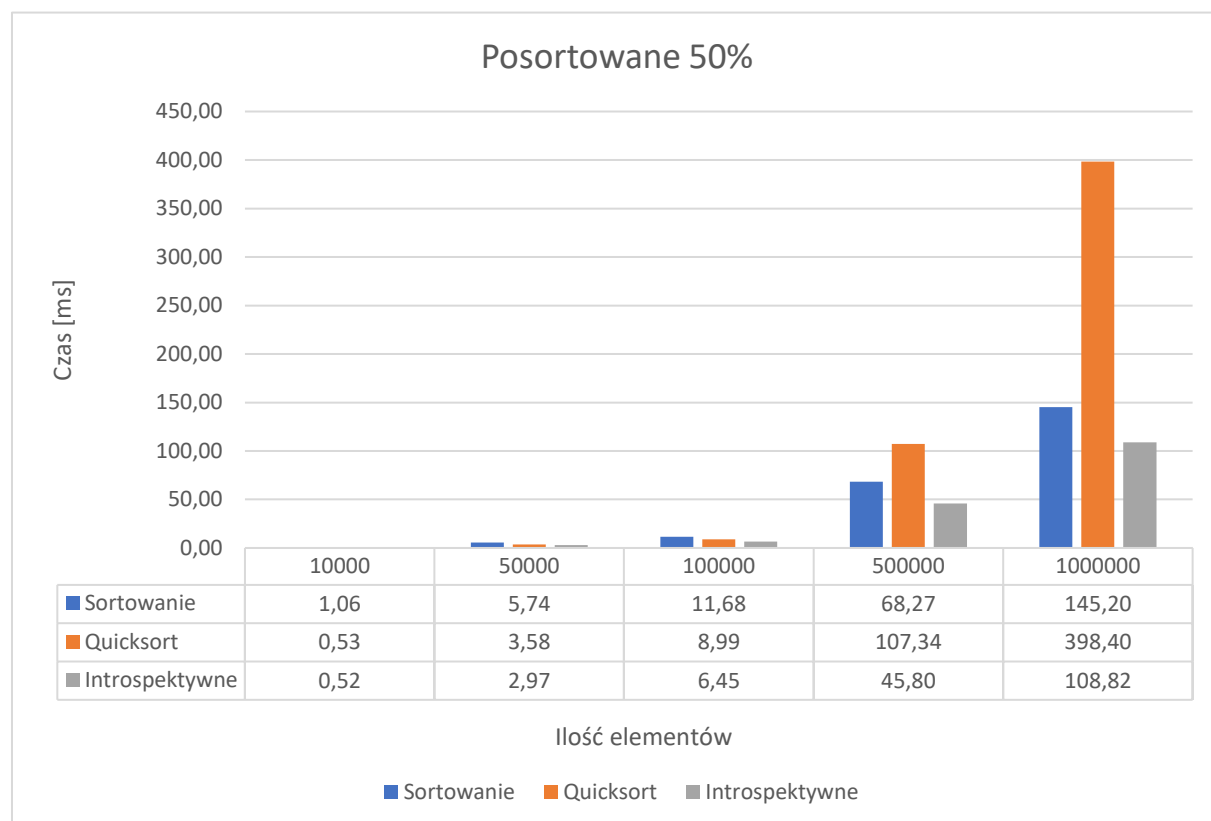
2. Otrzymane wyniki



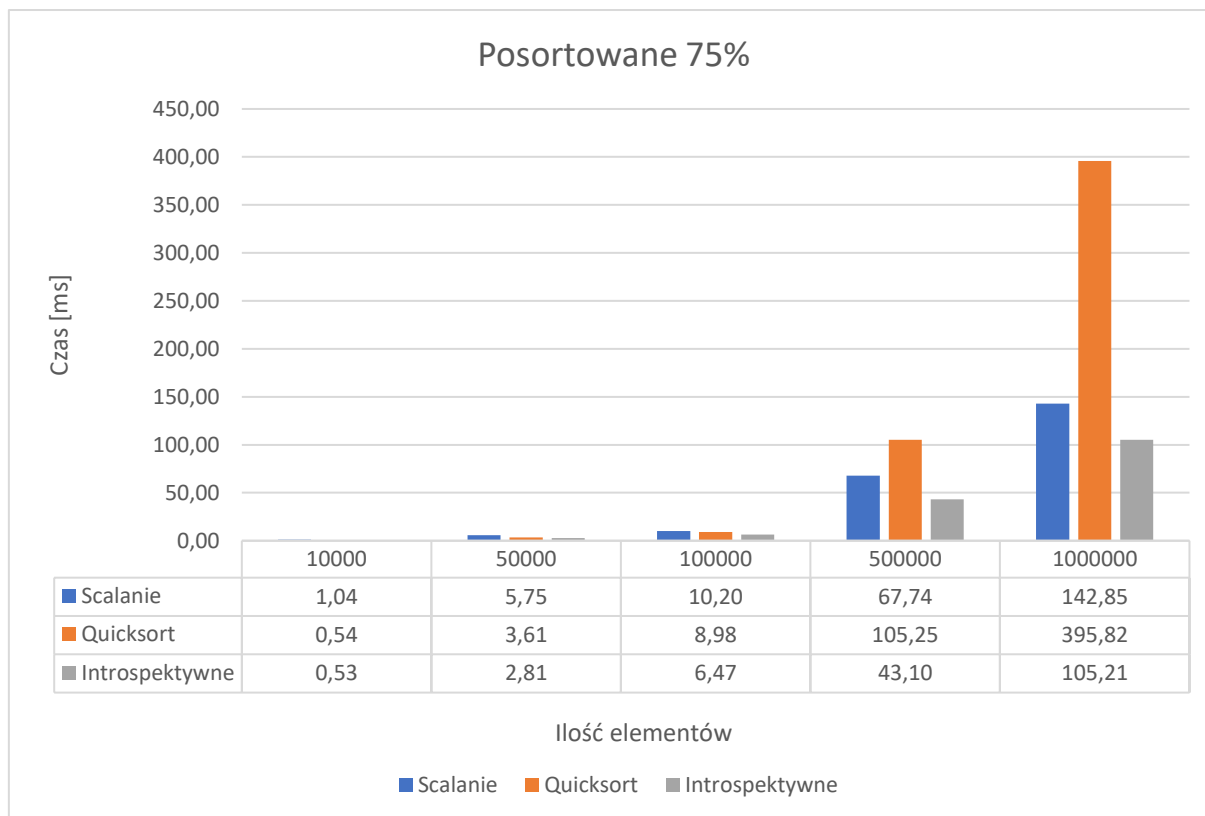
Wykres 1: Wyniki sortowania dla wszystkich elementów losowo



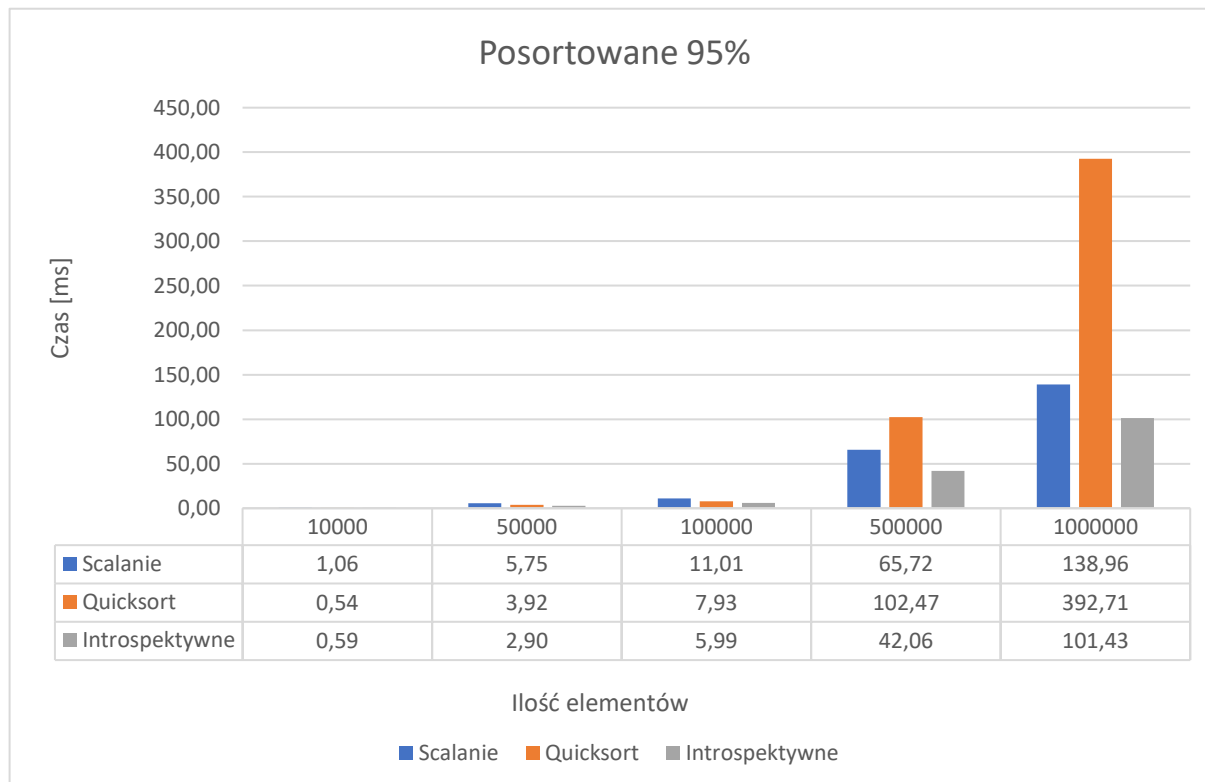
Wykres 2: Wyniki sortowania dla elementów posortowanych w 25%



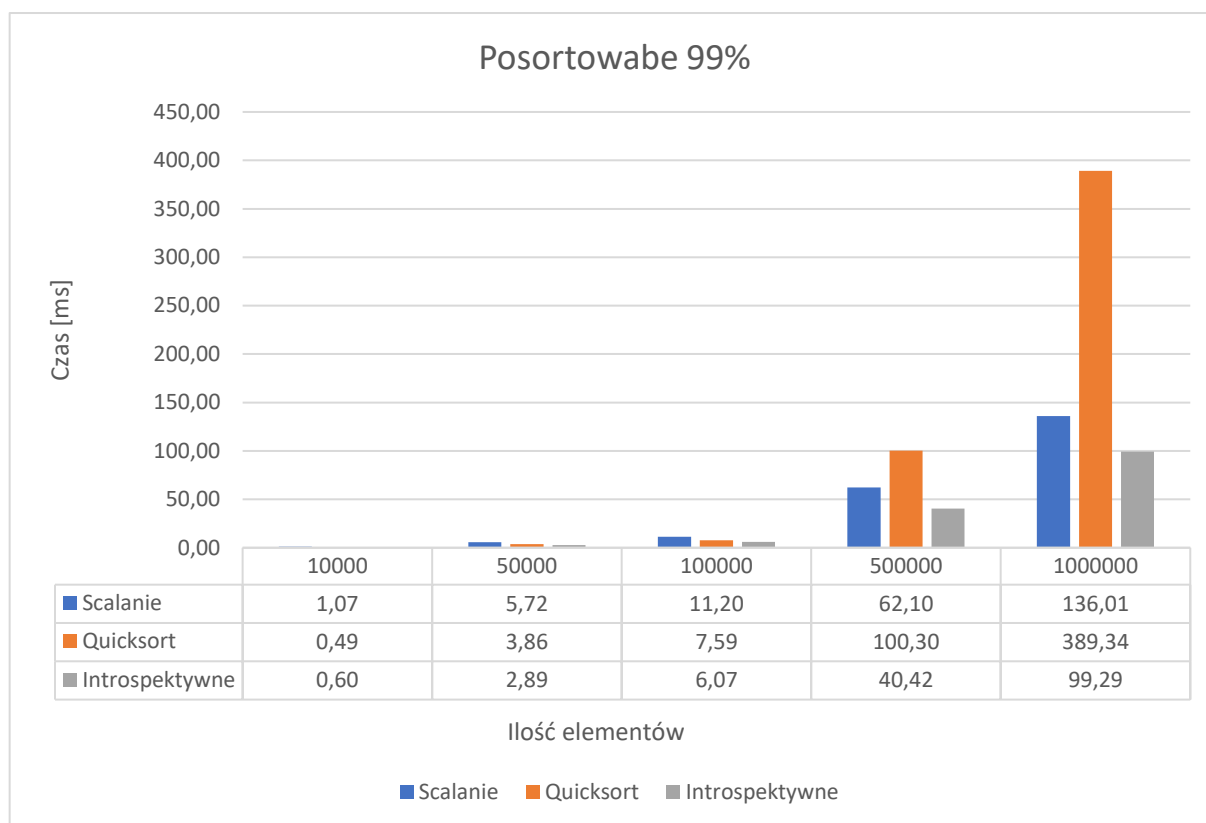
Wykres 3: Wyniki sortowania dla elementów posortowanych w 50%



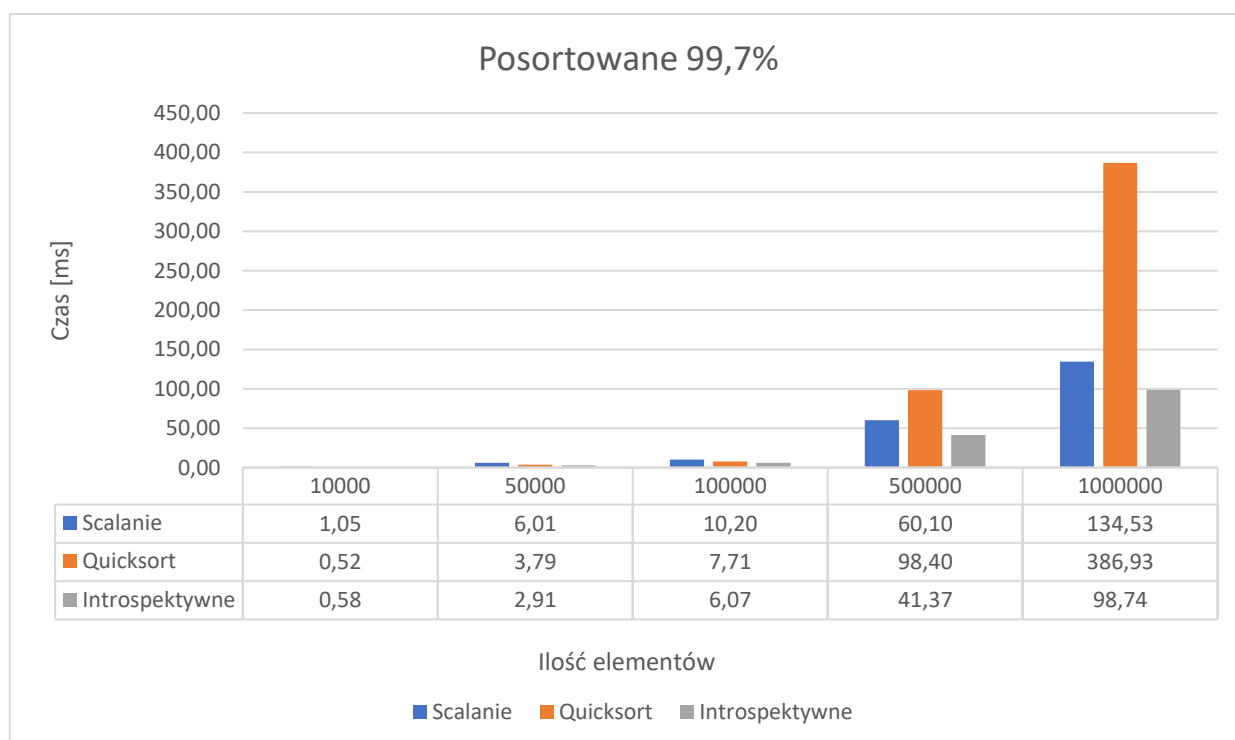
Wykres 4: Wyniki sortowania dla elementów posortowanych w 75%



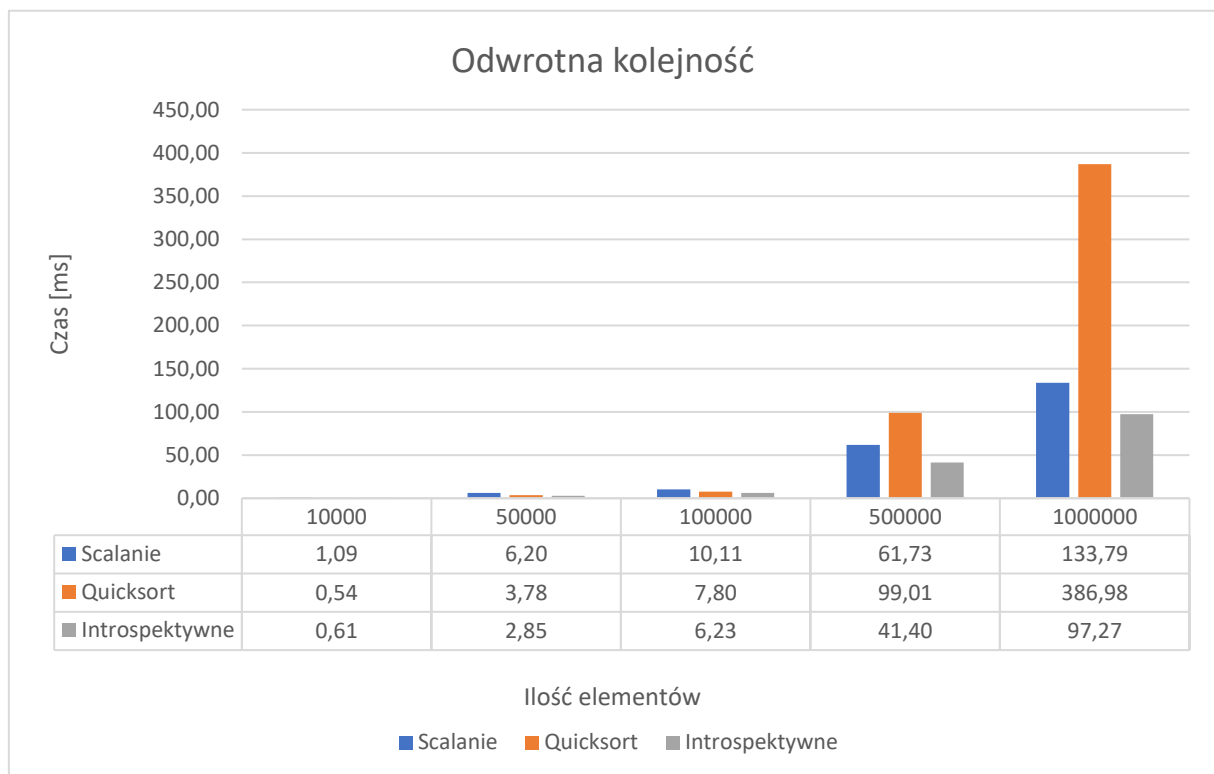
Wykres 5: Wyniki sortowania dla elementów posortowanych w 95%



Wykres 6: Wyniki sortowania dla elementów posortowanych w 99%



Wykres 7: Wyniki sortowania dla elementów posortowanych w 99,7%



Wykres 8: Wyniki sortowania dla elementów posortowanych w odwrotnej kolejności

3. Wnioski

Z wyników obserwujemy, że sortowanie introspektywne jest najszybsze. Jest to spowodowane tym, że jest to metoda hybrydowa, będąca połączeniem sortowania szybkiego oraz sortowania przez kopcowanie. Metoda quicksort wypadła najgorzej, może to być spowodowane złożonością obliczeń w najgorszym przypadku, która wynosi $O(n^2)$. Natomiast zwiększanie liczby posortowanych już elementów skraca czas sortowania.