

PROJEKTOWANIE ALGORYTMÓW I METOD SZTUCZNEJ INTELIGENCJI

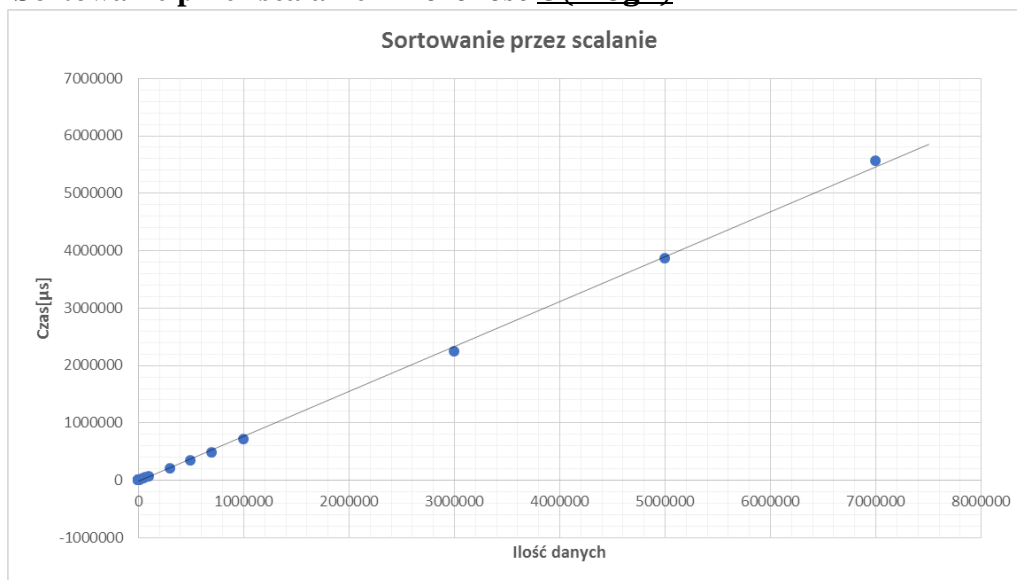
IMIĘ I NAZWISKO	NUMER INDEKSU	TERMIN
Kacper Połatajko	241603	Śr. 15 ¹⁵ – 16 ⁵⁵
TEMAT: Projekt 2 – Algorytmy sortowania		

1. Cel ćwiczenia

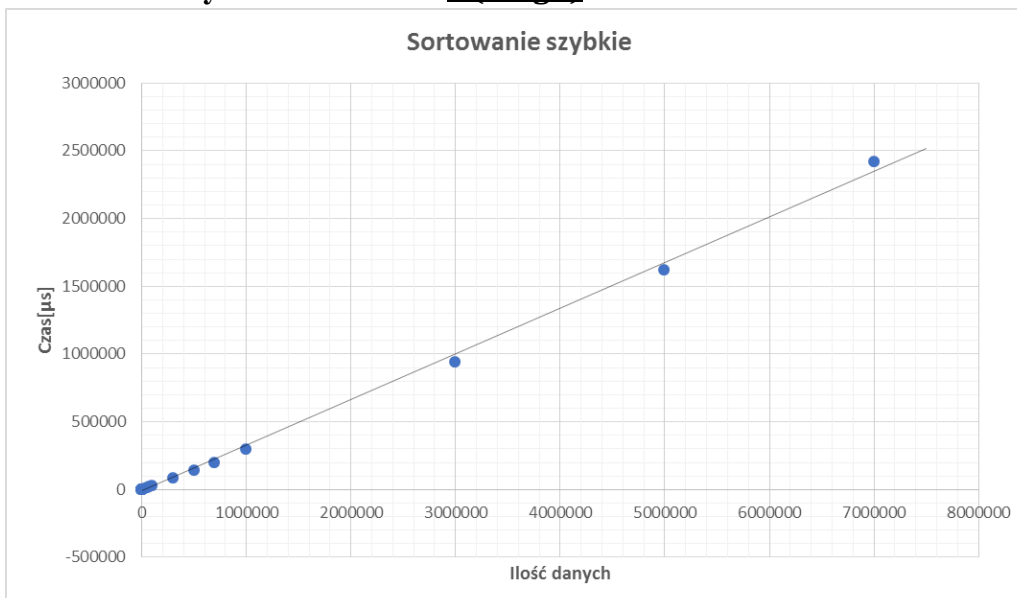
Zaimplementowanie trzech algorytmów sortowania dwa o złożoności obliczeniowej nie większej niż $O(n \log n)$ oraz jeden o dowolnej złożoności np. $O(n^2)$. Następnie sprawdzenie ich wydajności dla różnych rozmiarów problemów (dla różnej ilości danych wejściowych). Czas zliczany w μs .

2. Wynik działania sortowań

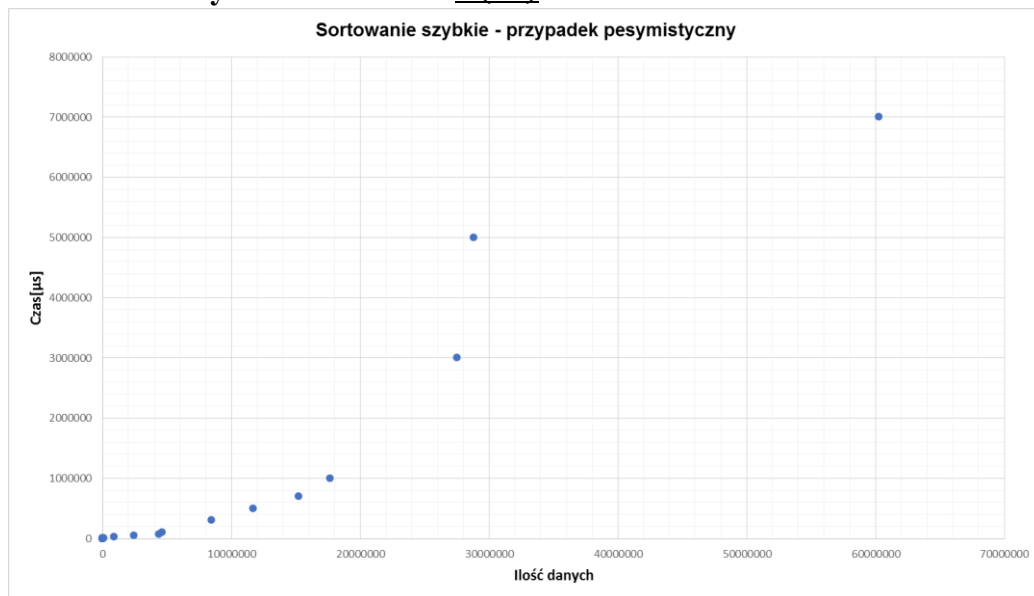
a. Sortowanie przez scalanie – złożoność $O(n \log n)$



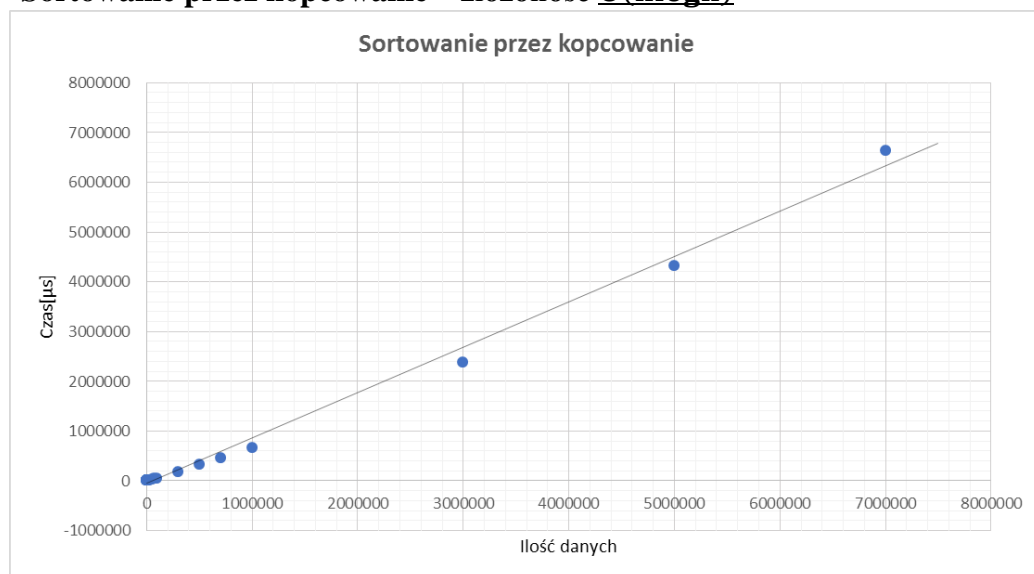
b. Sortowanie szybkie – złożoność $O(n \log n)$



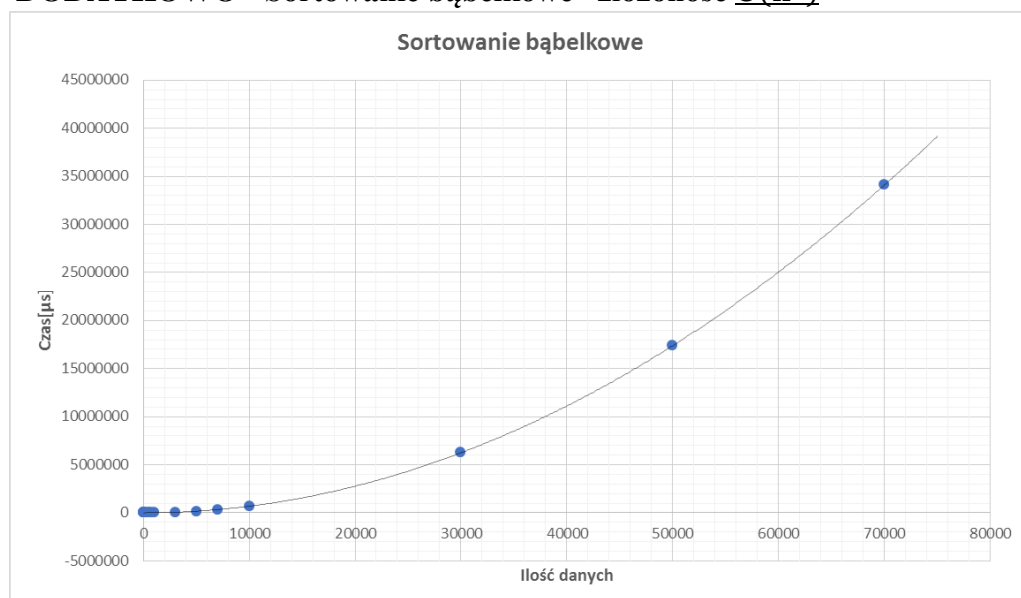
c. Sortowanie szybkie – złożoność $O(n^2)$



d. Sortowanie przez kopcowanie – złożoność $O(n \log n)$



e. DODATKOWO - Sortowanie bąbelkowe – złożoność $O(n^2)$



3. Wnioski

Wynikiem zastosowania algorytmów sortowania przez scalanie, przez kopcowanie i szybkiego (przypadek niepesymistyczny) są wykresy bardzo zbliżone do wykresu funkcji $n \log n$, tak jak tego oczekiwaliśmy.

Przypadek pesymistyczny dla algorytmu sortowania szybkiego zakładał sortowanie tablic wypełnionych w $3/4$ pojedynczą liczbą, zaś do pozostałej części zostały losowo przypisane liczby z zakresu od 0 do 100000. Poza pojedynczym wynikiem pomiaru (błąd może tkwić nie w kodzie czy algorytmie sortowania, a w mocy obliczeniowej laptopa) charakterystyka wyjściowa jest zbliżona do wykresu funkcji n^2 .

Dodatkowo został przedstawiony algorytm sortowania bąbelkowego o złożoności obliczeniowej $O(n^2)$ dla porównania z pesymistycznym przypadkiem sortowania szybkiego i dla pokazania różnicy w wykresach dla algorytmów o złożoności $O(n \log n)$.