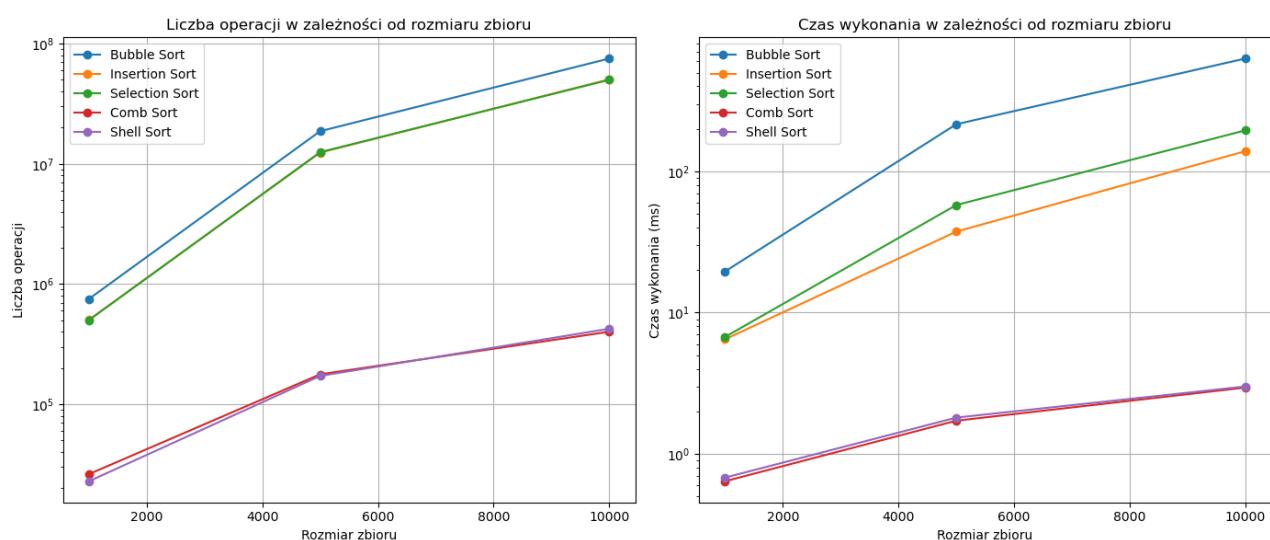


Proste Algorytmy Sortujące

Algorytmy:

- Sortowanie bąbelkowe (Bubble Sort)
- Sortowanie przez wstawianie (Insertion Sort)
- Sortowanie przez selekcję (Selection Sort)
- Sortowanie grzebieniowe (Comb Sort)
- Sortowanie Shella (Shell Sort)

Wyniki:



Analiza Wyników

1. Który algorytm okazał się najefektywniejszy dla dużych zbiorów danych?

Comb Sort oraz Shell Sort są zdecydowanie najefektywniejsze. Dla $n = 10\ 000$ wykonują się w czasie około 3 ms, podczas gdy Bubble Sort potrzebuje aż 628 ms.

2. Jakie różnice widać między algorytmami dla małych a dużych zbiorów?

Dla małych zbiorów danych ($n = 1\ 000$) różnice czasowe są stosunkowo niewielkie i mieszczą się w przedziale 0,6–19 ms. Dla dużych zbiorów różnice rosną dramatycznie, algorytmy o złożoności $O(n^2)$, takie jak Bubble Sort, Insertion Sort i Selection Sort, są około 200 razy wolniejsze niż Comb Sort i Shell Sort.

3. Czy algorytm z większą złożonością teoretyczną może być praktycznie szybszy?

Tak. Comb Sort, który ma średnią złożoność $O(n^2 / 2^p)$, jest w praktyce znacznie szybszy niż proste algorytmy $O(n^2)$. Dla $n = 10\ 000$ Comb Sort osiąga czas około 2,9 ms, podczas gdy Bubble Sort aż 628 ms. Pokazuje to, że stałe współczynniki oraz optymalizacje implementacyjne mają ogromne znaczenie w praktyce.