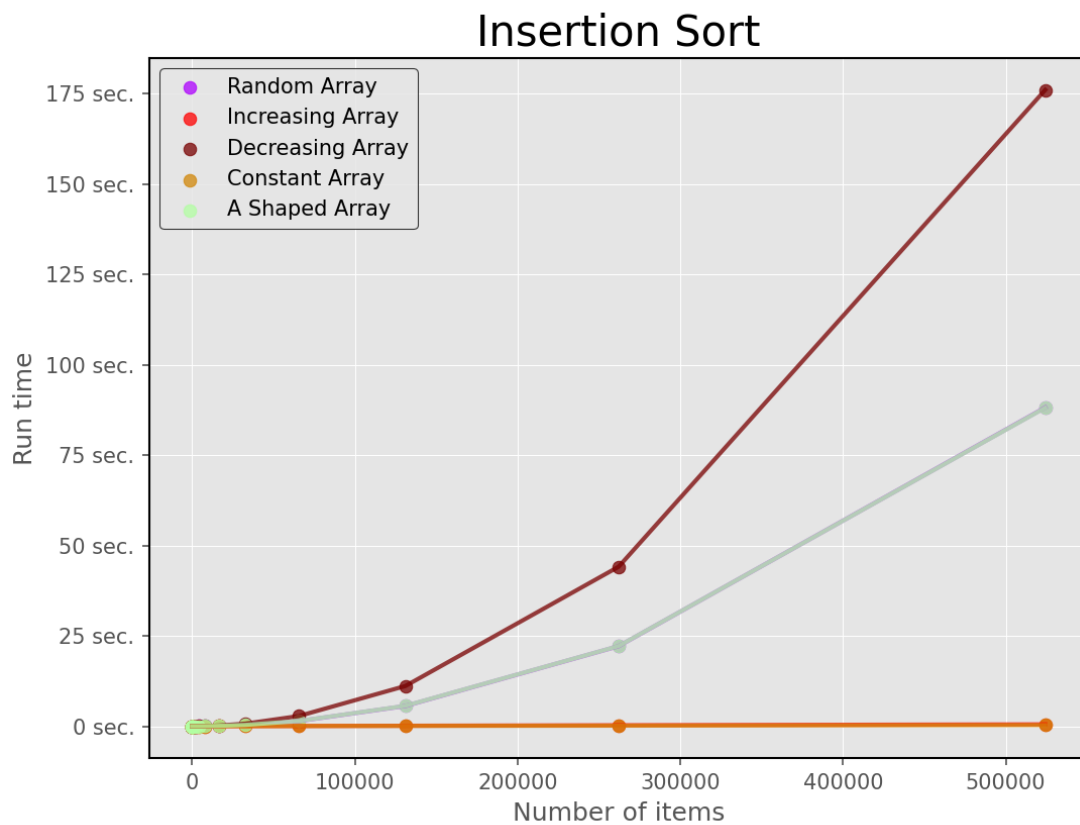


1. Insertion Sort:

Złożoność obliczeniowa		
Minimalna	Maksymalna	Średnia
$O(n)$	$O(n^2)$	$O(n^2)$

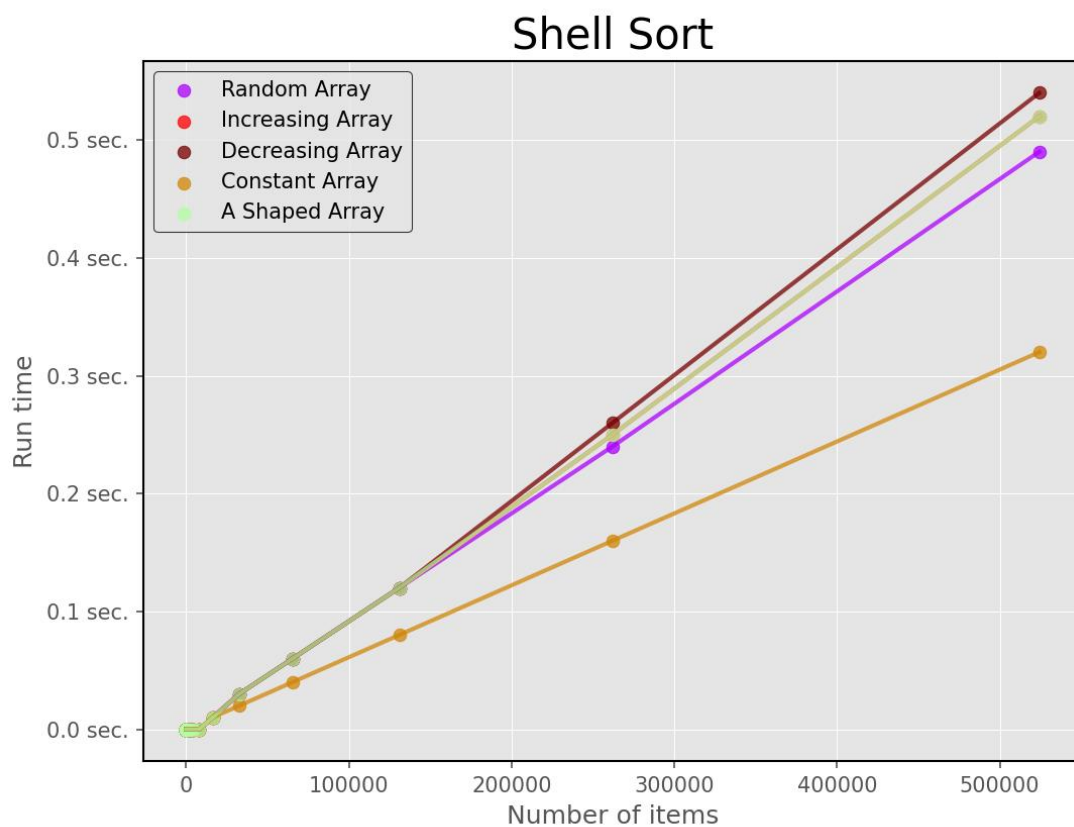
Insertion Sort sortuje elementy poprzez iteracyjne "wstawianie" każdego elementu na odpowiednie miejsce w posortowanej już części tablicy.



2. Shell Sort:

Złożoność obliczeniowa		
Minimalna	Maksymalna	Średnia
$O(n \log n)$	$O(n^2)$	$O(n^{\frac{4}{3}})$

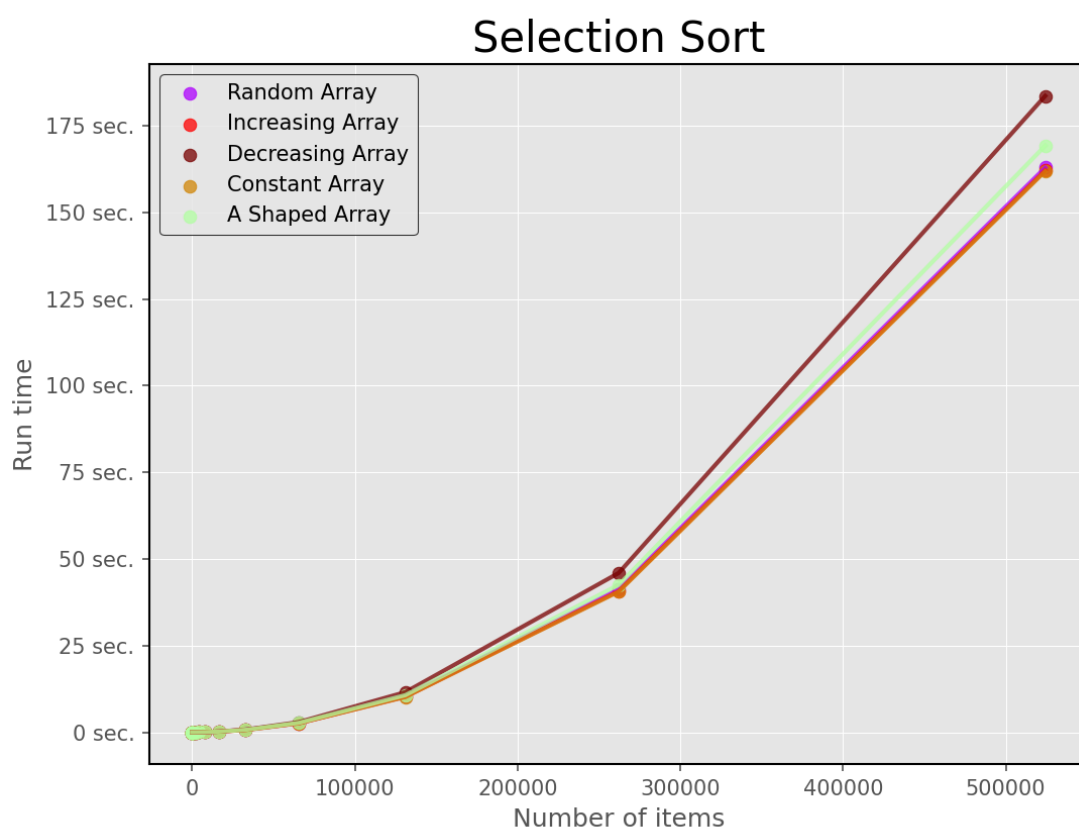
Shell Sort jest ulepszoną wersją Insertion Sort, która dzieli listę na mniejsze podlisty, a następnie sortuje je przy użyciu Insertion Sort.



3. Selection Sort:

Złożoność obliczeniowa		
Minimalna	Maksymalna	Średnia
$O(n^2)$	$O(n^2)$	$O(n^2)$

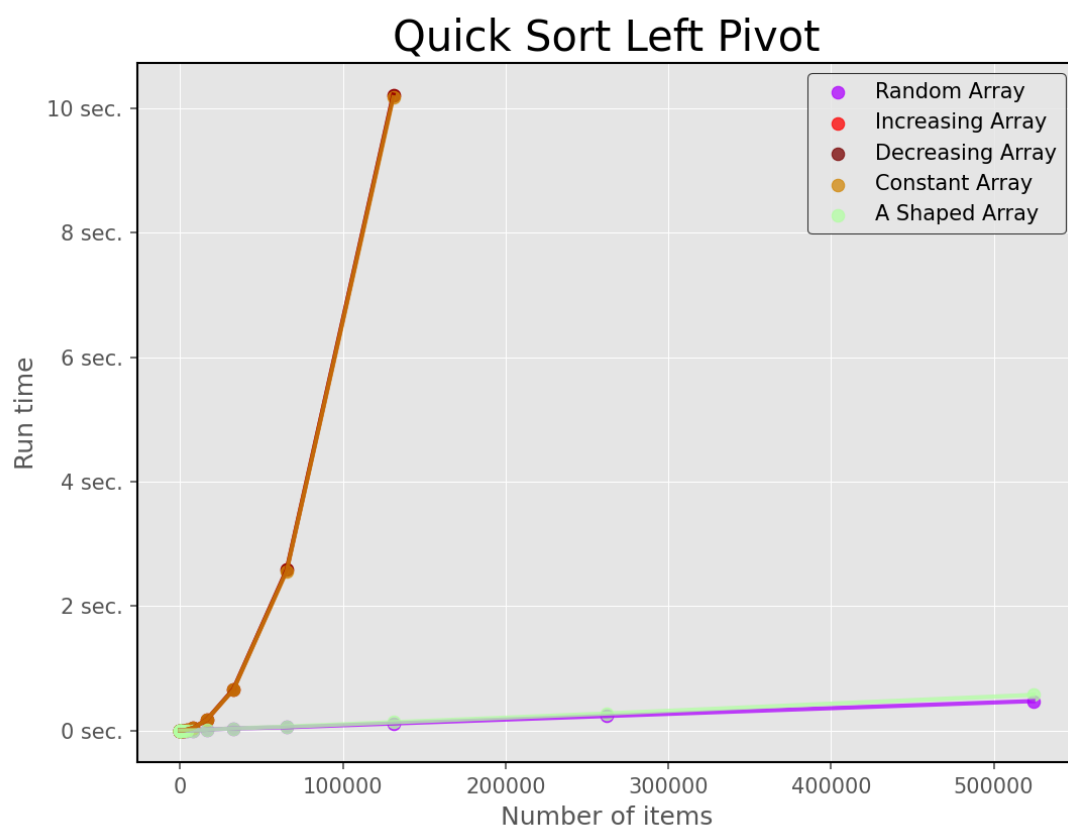
Selection Sort znajduje najmniejszy element w nieposortowanej części listy i zamienia go z pierwszym nieposortowanym elementem.



4. Quick Sort (z pivotem na lewo):

Złożoność obliczeniowa		
Minimalna	Maksymalna	Średnia
$O(n \log n)$	$O(n^2)$	$O(n \log n)$

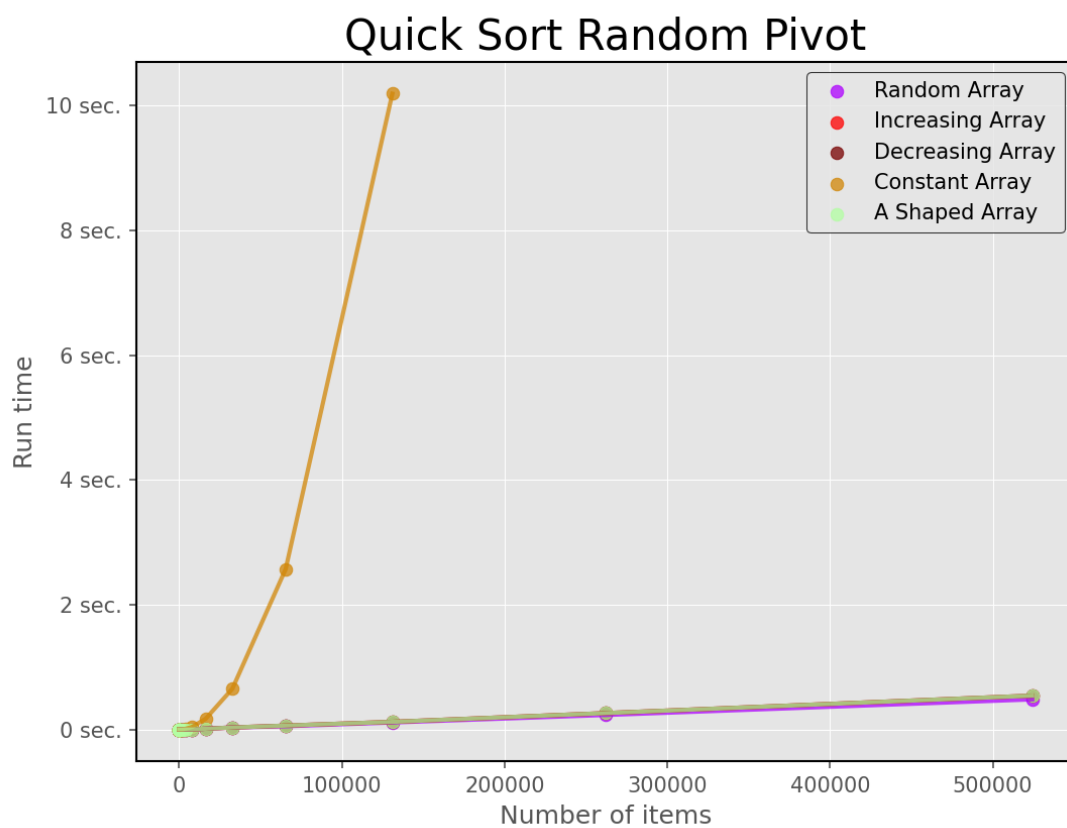
Quick Sort to algorytm sortowania dziel i zwyciężaj. Wybiera się element pivotowy, a następnie dzieli się listę na dwa podzbiory - mniejsze od pivota i większe od pivota. Następnie sortuje się rekurencyjnie oba podzbiory.



5. Quick Sort (z losowym pivotem):

Złożoność obliczeniowa		
Minimalna	Maksymalna	Średnia
$O(n \log n)$	$O(n^2)$	$O(n \log n)$

Ten algorytm działa tak samo jak Quick Sort z pivotem na lewo, z wyjątkiem tego, że pivot jest losowo wybierany spośród elementów listy.



6. Heap Sort:

Złożoność obliczeniowa		
Minimalna	Maksymalna	Średnia
$O(n \log n)$	$O(n \log n)$	$O(n \log n)$

Heap Sort tworzy maksymalne drzewo kopcowe z danych, a następnie iteracyjnie usuwa największy element z korzenia kopca i przywraca kopiec do jego własności, tworząc posortowaną listę.

