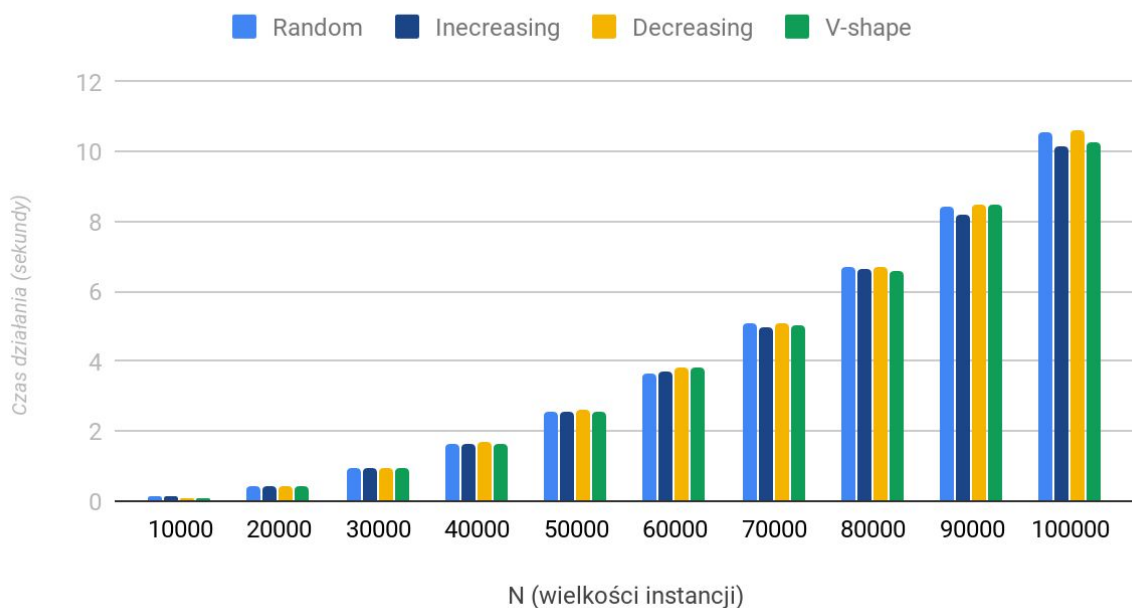


Selection sort

N (wielkości instancji)	V-shape	Random	Increasing	Decreasing
10000	0,1	0,154	0,115	0,1
20000	0,417	0,451	0,402	0,417
30000	0,918	0,933	0,949	0,924
40000	1,651	1,62	1,62	1,667
50000	2,561	2,56	2,545	2,609
60000	3,809	3,641	3,694	3,846
70000	5,014	5,114	4,976	5,098
80000	6,565	6,67	6,618	6,703
90000	8,461	8,439	8,207	8,454
100000	10,274	10,529	10,159	10,598

Selection Sort



Czas działania algorytmu jest jednolity dla różnego typu danych wejściowych, rośnie on zaś parabolicznie w stosunku do ich rozmiaru.

Złożoność obliczeniowa jest identyczna dla wszystkich przypadków (wynosi ona $O(n^2)$), nie jest możliwe więc wyodrębnienie przypadku pesymistycznego (o większej złożoności względem innych przypadków/największym czasie działania w proporcji do ilości danych wejściowych), ani przypadku optymistycznego (o najmniejszej możliwej złożoności/najmniejszym możliwym czasie działania względem ilości danych wejściowych).

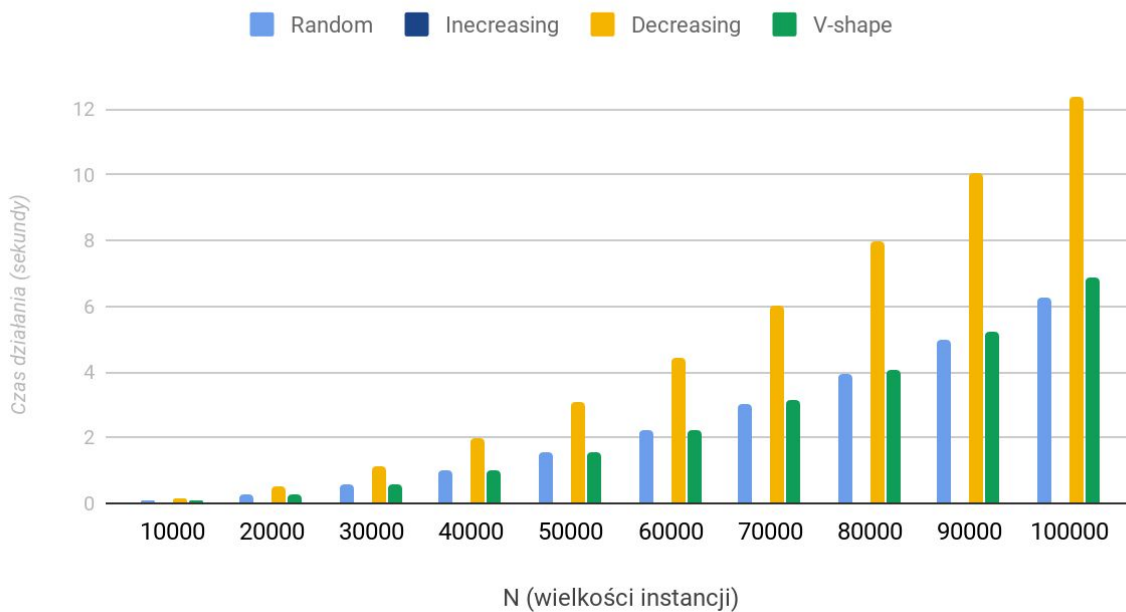
Random - złożoność $O(n^2)$,

Increasing - złożoność $O(n^2)$,
Decreasing - złożoność $O(n^2)$,
V-shape - złożoność $O(n^2)$,

Insertion Sort

N (wielkości instancji)	Random	Increasing	Decreasing	V-shape
10000	0,069	0	0,123	0,084
20000	0,254	0	0,495	0,279
30000	0,548	0	1,109	0,555
40000	0,987	0	1,968	0,987
50000	1,535	0	3,093	1,535
60000	2,222	0	4,443	2,222
70000	3,046	0	6,039	3,146
80000	3,958	0	7,968	4,089
90000	4,968	0	10,044	5,199
100000	6,248	0	12,371	6,855

Insertion Sort



Czas działania jest najszybszy dla danych już posortowanych, których złożoność obliczeniowa wynosi $O(n)$.

Przypadkiem pesymistycznym jest zaś sekwencja danych ułożonych malejąco, dla których złożoność obliczeniowa wynosi $O(n^2)$.

Random - złożoność $O(n)$ lub $O(n^2)$,

Increasing - złożoność $O(n)$,

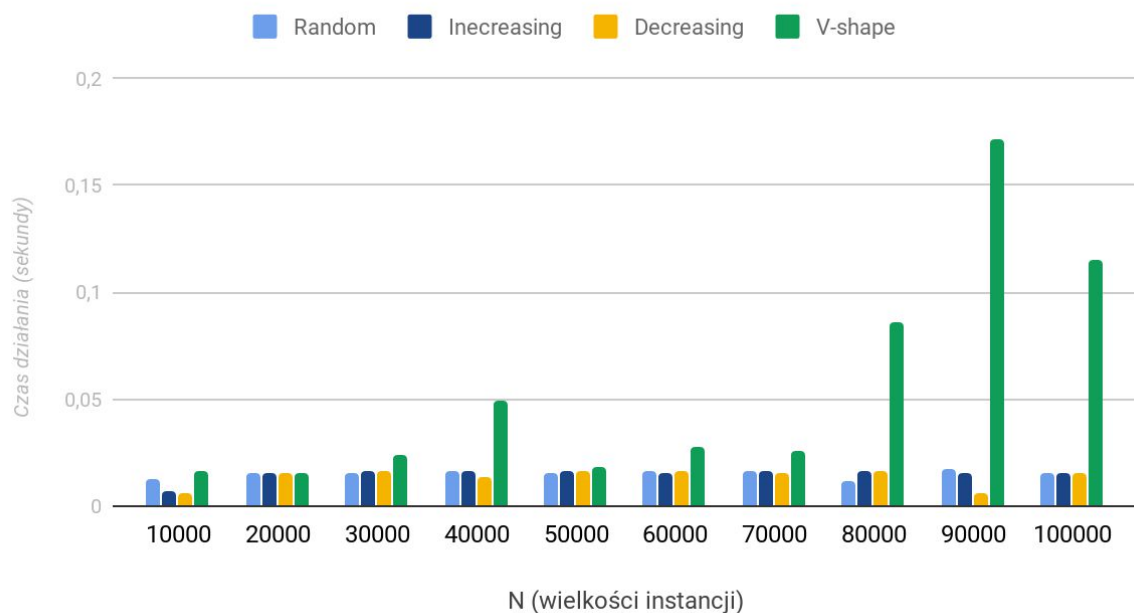
Decreasing - złożoność $O(n^2)$

v-shape - złożoność $O(n)$

Quick Sort

N (wielkości instancji)	Random	Increasing	Decreasing	V-shape
10000	0,001	0,0001	0,001	0,003
20000	0,002	0,001	0,001	0,006
30000	0,004	0,002	0,002	0,025
40000	0,006	0,002	0,002	0,058
50000	0,007	0,002	0,002	0,027
60000	0,008	0,003	0,002	0,034
70000	0,010	0,003	0,003	0,027
80000	0,012	0,004	0,003	0,108
90000	0,013	0,004	0,004	0,207
100000	0,014	0,005	0,005	0,132

Quick Sort



Algorytm quicksort działa w tempie logarytmicznym dla większości typów danych wejściowych ($O(n \log n)$).

Występuje jednak przypadek pesymistyczny (w tym wypadku jest to V-shape), dla którego złożoność obliczeniowa równa jest $O(n^2)$.

Random - złożoność $O(n \log n)$

Increasing - złożoność $O(n \log n)$

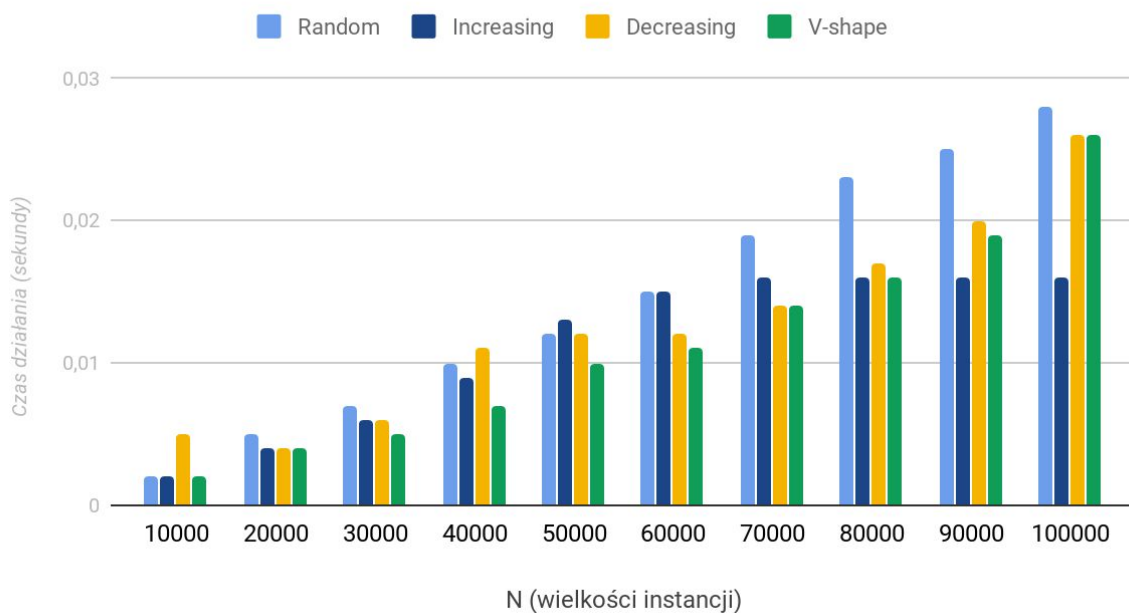
Decreasing - złożoność $O(n \log n)$

V-shape - złożoność $O(n^2)$

Heap Sort

N (wielkości instancji)	Random	Increasing	Decreasing	V-shape
10000	0,002	0,002	0,005	0,002
20000	0,005	0,004	0,004	0,004
30000	0,007	0,006	0,006	0,005
40000	0,010	0,009	0,011	0,007
50000	0,012	0,013	0,012	0,01
60000	0,015	0,015	0,012	0,011
70000	0,019	0,016	0,014	0,014
80000	0,023	0,018	0,017	0,016
90000	0,025	0,020	0,02	0,019
100000	0,028	0,022	0,026	0,026

Heap Sort



Czas działania jest największy w przypadku już uporządkowanych danych.

Przypadkiem pesymistycznym są zaś dane wprowadzane losowo, których złożoność obliczeniowa wynosi $O(n \log n)$. Operuje on wolniej w porównaniu do algorytmu quicksort i jest algorytmem o mniejszej stabilności, jednak posiada znacznie mniejszą złożoność w przypadku pesymistycznym.

Random - złożoność $O(n \log n)$

Increasing - złożoność $O(n \log n)$

Decreasing - złożoność $O(n \log n)$

V-shape - złożoność $O(n \log n)$