

# Sprawozdanie

Algorytmy i struktury danych

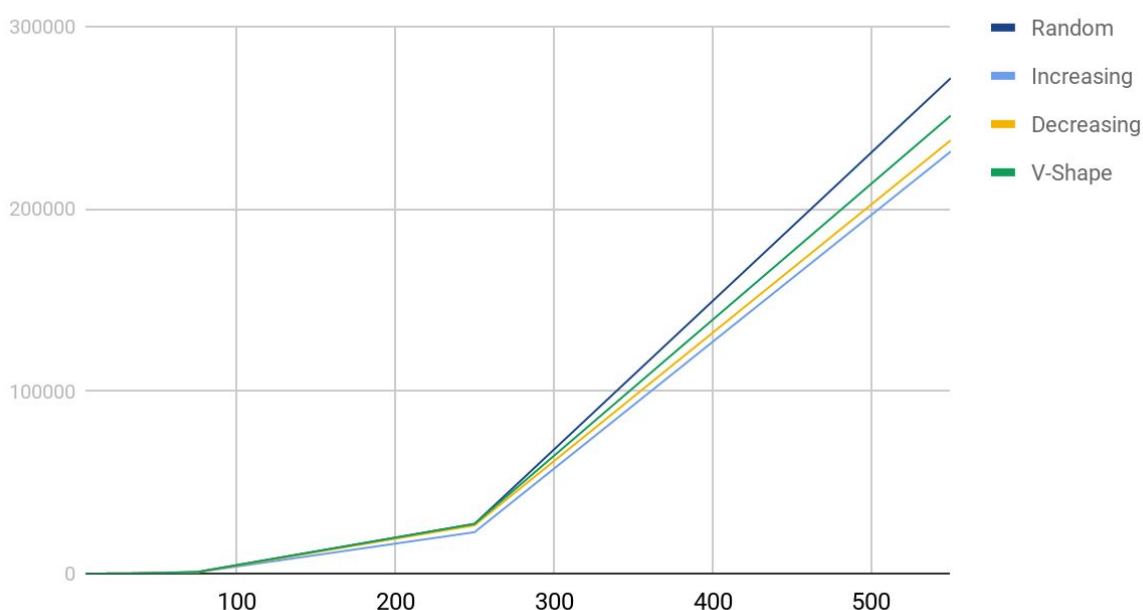
**Xenia Pietrzyk**  
**nr indeksu: 146409**  
**grupa 4B**

# Selection Sort

Selection Sort	5	10	15	20	25	30	35	75	250	550
Random	0.000002	0.000004	0.000008	0.000019	0.000033	0.000056	0.000087	0.000797	0.027225	0.271686
Increasing	0.000001	0.000003	0.000007	0.000017	0.000028	0.000053	0.000081	0.000669	0.022676	0.231464
Decreasing	0.000002	0.000003	0.000008	0.000016	0.000030	0.000051	0.000082	0.000681	0.026439	0.237630
V-Shape	0.000001	0.000004	0.000008	0.000017	0.000030	0.000053	0.000090	0.000770	0.027222	0.251119

Tab.1. Tabela obrazująca czas realizacji sortowania dla tablic o różnych wielkościach przez algorytm Selection Sort.

## Selection Sort



Wykres obrazujący dane z powyższej tabeli Tab.1.

**Wnioski:** Algorytm selection sort wykazuje liniową tendencję wzrostu czasu potrzebnego do wykonania sortowania w stosunku do ilości elementów znajdujących się w sortowanej tablicy.

Różnice pomiędzy czasem potrzebnym do sortowania dla rozłożenia liczb w tablicy: a) losowego, b) wzrastającego, c) malejącego, d) v-kształtnego są niewielkie.

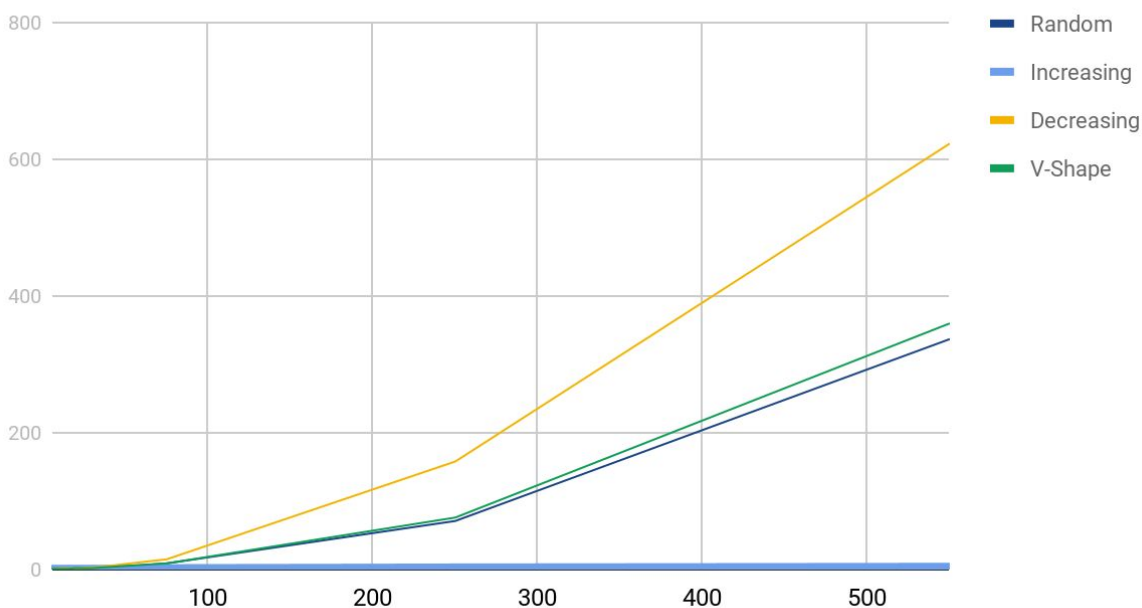
Najbardziej korzystne rozłożenie to wariant wzrastający, najmniej korzystny - losowy.

# Insertion Sort

Insertion Sort	5	10	15	20	25	30	35	75	250	550
Random	0.000001	0.000001	0.000001	0.000001	0.000002	0.000002	0.000002	0.000009	0.000071	0.000337
Increasing	0.000001	0.000001	0.000001	0.000001	0.000001	0.000001	0.000001	0.000002	0.000003	0.000004
Decreasing	0.000001	0.000002	0.000002	0.000002	0.000003	0.000003	0.000004	0.000015	0.000158	0.000623
V-Shape	0.000001	0.000001	0.000001	0.000002	0.000002	0.000002	0.000003	0.000009	0.000076	0.000360

Tab.2. Tabela obrazująca czas realizacji sortowania dla tablic o różnych wielkościach przez algorytm Insertion Sort.

## Insertion Sort



Wykres obrazujący dane z powyższej tabeli Tab.2.

**Wnioski:** Algorytm insertion sort wykazuje większe rozproszenie poszczególnych wyników w porównaniu do sortowania selection sort.

Różnice pomiędzy czasem potrzebnym do sortowania dla rozłożenia liczb w tablicy: a) losowego, b) wzrastającego, c) malejącego, d) v-kształtnego są widocznie większe, niż te zaobserwowane w algorytmie Selection Sort. Szczególną uwagę zwraca wynik dla rozłożenia malejącego - jest on najmniej korzystnie

wypadającym rozłożeniem, oraz dla rozłożenia wzrastającego - jest ono najbardziej korzystnie wypadającym w powyższym wykresie.

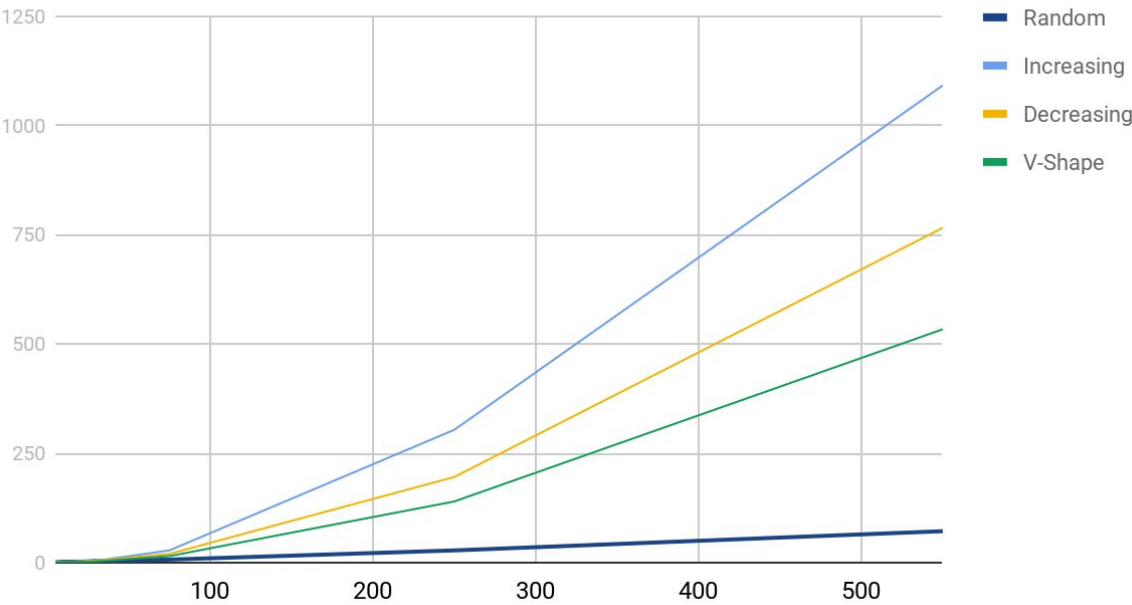
## Quick Sort

dla pivot definiowanego przez ostatni element tablicy

Insertion Sort	5	10	15	20	25	30	35	75	250	550
Random	0.000001	0.000002	0.000002	0.000002	0.000003	0.000004	0.000004	0.000007	0.000028	0.000072
Increasing	0.000001	0.000002	0.000002	0.000003	0.000004	0.000006	0.000007	0.000028	0.000304	0.001092
Decreasing	0.000001	0.000001	0.000002	0.000003	0.000004	0.000004	0.000006	0.000020	0.000196	0.000766
V-Shape	0.000001	0.000002	0.000002	0.000002	0.000002	0.000003	0.000004	0.000015	0.000140	0.000534

Tab.3. Tabela obrazująca czas realizacji sortowania dla tablic o różnych wielkościach przez algorytm Quick Sort.

## Quick Sort



Wykres obrazujący dane z powyższej tabeli Tab.3.

**Wnioski:** Algorytm quick sort wykazuje największe rozproszenie poszczególnych wyników dla badanych próbek.

Różnice pomiędzy czasem potrzebnym do sortowania dla rozłożenia liczb w tablicy: a) losowego, b) wzrastającego, c) malejącego, d) v-kształtnego są widocznie największe spośród zaobserwowanych w poprzednich algorytmach. Układ najkorzystniejszy to ten o losowym rozłożeniu liczb w badanych tablicach, natomiast najmniej korzystny to ten o wzrastającym rozłożeniu liczb.

## Porównanie zbiorcze

Random	Selection sort	Insertion Sort	Quick Sort
5	0.000002	0.000001	0.000001
10	0.000004	0.000001	0.000002
15	0.000008	0.000001	0.000002
20	0.000019	0.000001	0.000002
25	0.000033	0.000002	0.000003
30	0.000056	0.000002	0.000004
35	0.000087	0.000002	0.000004
75	0.000797	0.000009	0.000007
250	0.027225	0.000071	0.000028
550	0.271686	0.000337	0.000072

Tab.4. Tabela obrazująca czas realizacji sortowania dla tablic o różnych wielkościach przez zbadane algorytmy dla rozłożenia losowego liczb w tablicach.

Increasing	Selection sort	Insertion Sort	Quick Sort
5	0.000001	0.000001	0.000001
10	0.000003	0.000001	0.000002
15	0.000007	0.000001	0.000002
20	0.000017	0.000001	0.000003
25	0.000028	0.000001	0.000004
30	0.000053	0.000001	0.000006
35	0.000081	0.000001	0.000007
75	0.000669	0.000002	0.000028
250	0.022676	0.000003	0.000304
550	0.231464	0.000004	0.001092

Tab.5. Tabela obrazująca czas realizacji sortowania dla tablic o różnych wielkościach przez zbadane algorytmy dla rozłożenia  
wzrastającego liczb w tablicach.

Decreasing	Selection sort	Insertion Sort	Quick Sort
5	0.000002	0.000001	0.000001
10	0.000003	0.000002	0.000001
15	0.000008	0.000002	0.000002
20	0.000016	0.000002	0.000003
25	0.000030	0.000003	0.000004
30	0.000051	0.000003	0.000004
35	0.000082	0.000004	0.000006
75	0.000681	0.000015	0.000020
250	0.026439	0.000158	0.000196
550	0.237630	0.000623	0.000766

Tab.6. Tabela obrazująca czas realizacji sortowania dla tablic o różnych wielkościach przez zbadane algorytmy dla rozłożenia  
malejącego liczb w tablicach.

V-Shape	Selection sort	Insertion Sort	Quick Sort
5	0.000001	0.000001	0.000001
10	0.000004	0.000001	0.000002
15	0.000008	0.000001	0.000002
20	0.000017	0.000002	0.000002
25	0.000030	0.000002	0.000002
30	0.000053	0.000002	0.000003
35	0.000090	0.000003	0.000004
75	0.000770	0.000009	0.000015
250	0.027222	0.000076	0.000140
550	0.251119	0.000360	0.000534

Tab.6. Tabela obrazująca czas realizacji sortowania dla tablic o różnych wielkościach przez zbadane algorytmy dla rozłożenia v-kształtnego liczb w tablicach.

**Wnioski:** Najmniej korzystnym czasem wykonywania obliczeń dla badanych próbek wykazał się algorytm selection sort. W każdym z badanych sposobów rozłożenia liczb w tablicy osiągał najgorsze wyniki.

W przypadku tablic, w których liczby ułożone były: a) wzrastająco, b) malejąco oraz c) v-kształtnie - najkorzystniejszym czasem wykonywania obliczeń dla badanych próbek wykazał się algorytm selection sort.

Istotnym zaobserwowanym faktem jest najlepsze działanie algorytmu quick sort dla badanych próbek w przypadku losowego rozłożenia liczb w tablicach. Osiągnął on znaczną przewagę czasową w porównaniu do algorytmu selection sort już na poziomie badania 15-elementowej tablicy. Różnica czasu wzrastała gwałtownie wraz ze wzrostem elementów w badanych tablicach.

Algorytm insertion sort wykazywał się podobnie korzystnym czasem potrzebnym do obliczeń do momentu badania tablicy 250-elementowej. Wówczas następuje duża rozbieżność pomiędzy czasem potrzebnym na posortowanie tablicy przez algorytm quick sort a insertion sort.

Algorytm quick sort jest najlepszym, spośród przebadanych do sortowania dużych tablic o losowym rozłożeniu liczb.

Algorytm insertion sort jest najlepszym spośród przebadanych do sortowania ltablic o v-kształtnym rozłożeniu liczb.

Algorytm selection sort nie wykazał podczas badań cech wyróżniających go pozytywnie na tle pozostałych algorytmów.