

Sprawozdanie  
Projektowanie algorytmów i metod sztucznej  
inteligencji  
Laboratorium 4

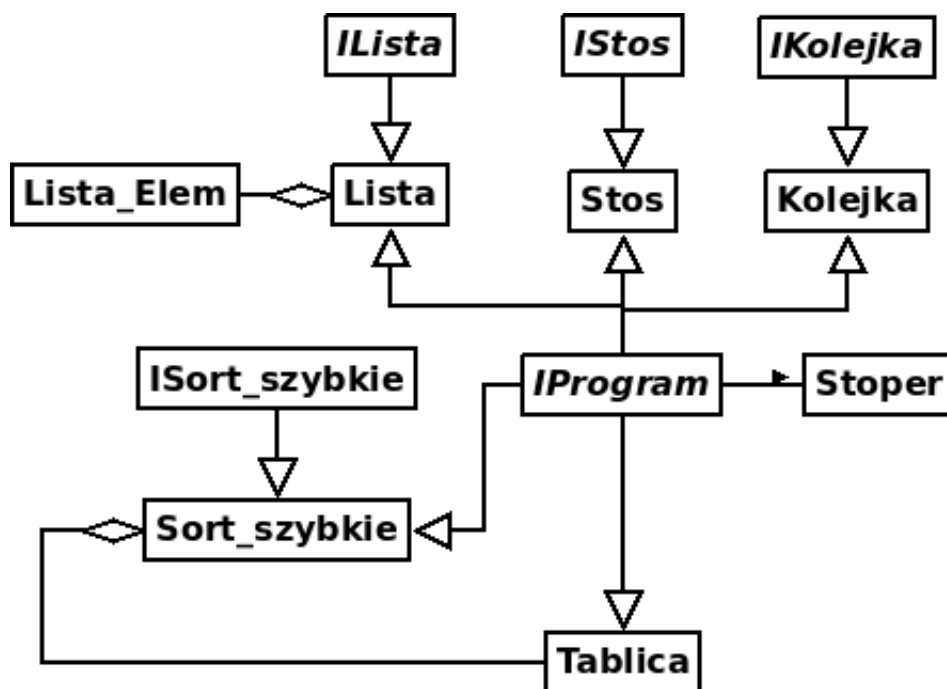
Michał Wieczorek  
226284

18 kwietnia 2017

# 1 Cel i przebieg labolatorium

Zapoznanie się z algorytmem sortowania szybkiego danych w tablicy. Budowa algorytmu szybkiego sortowania danych. Najgorsze i najlepsze przypadki działania algorytmu. Wpływ wyboru pivota na czas sortowania danych. Złożoność obliczeniowa algorytmu.

## 2 Testy programu



Rysunek 1: Diagram ilustrujący zaimplementowane interfejsy i klasy

### 2.1 Schemat ideowy

Do programu głównego dodano interfejs Sortowanie szybkie, którego metoda wirtualna nosi nazwę sortuj dane. Klasa Sortowanie szybkie dziedziczy bezpośrednio z wyżej wymienionego interfejsu oraz interfejsu głównego Program,

który zawiera w sobie metodę wykonaj program.

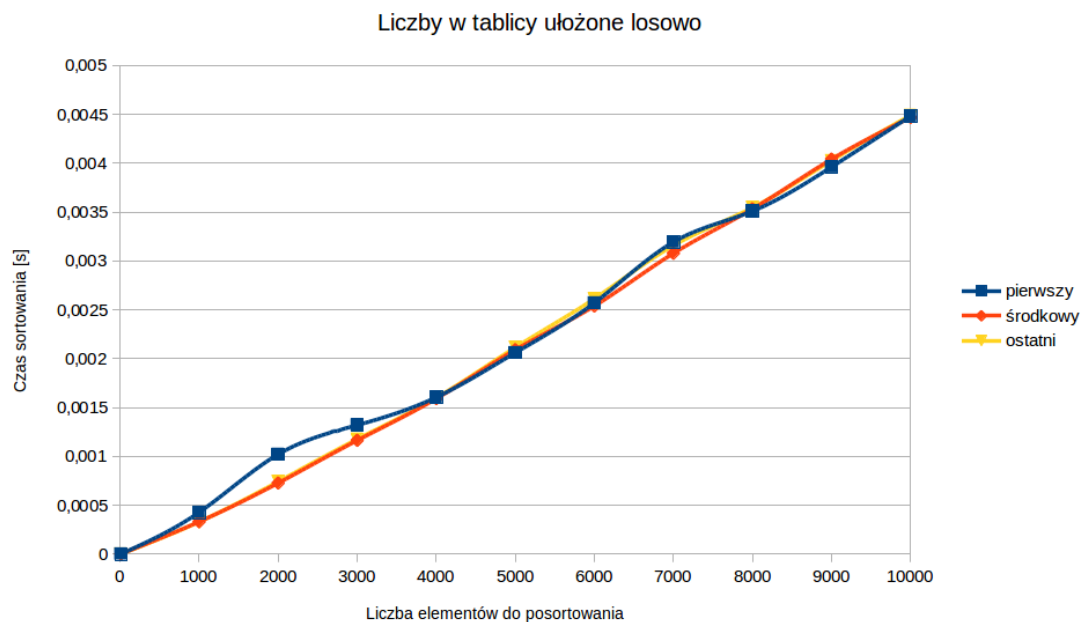
Sortowanie danych odbywa się na liczbach przechowywanych w Tablicy, którą tworzyliśmy na wcześniejszych zajęciach.

## 2.2 Działanie algorytmu

Algorytm sortowania pozwala na wybór ułożenia danych w tablicy (pseudo-losowe, rosnące, malejące) oraz na wybór elementu dzielącego (na początku, na środku, na końcu).

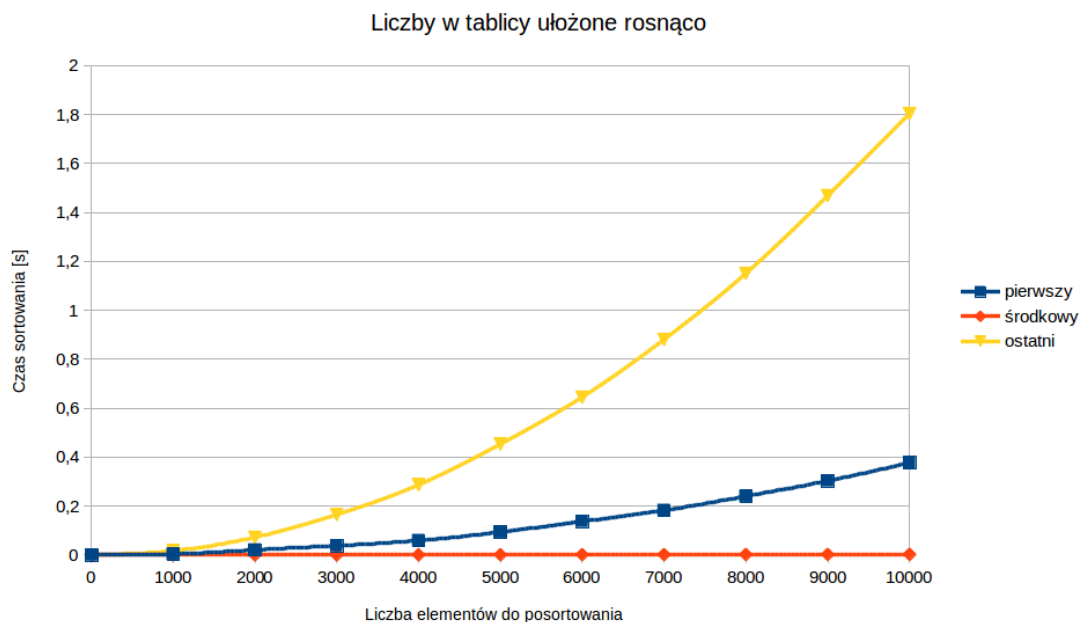
## 2.3 Pomiary

Pomiary zostały wykonane dla liczby elementów do posortowania: od 10 do 10000, co tysiąc elementów. Dla każdej liczby elementów wykonano po 50 pomiarów, a następnie policzono z nich średnią. Wyniki pomiarów zamieszczono na końcu dokumentu.

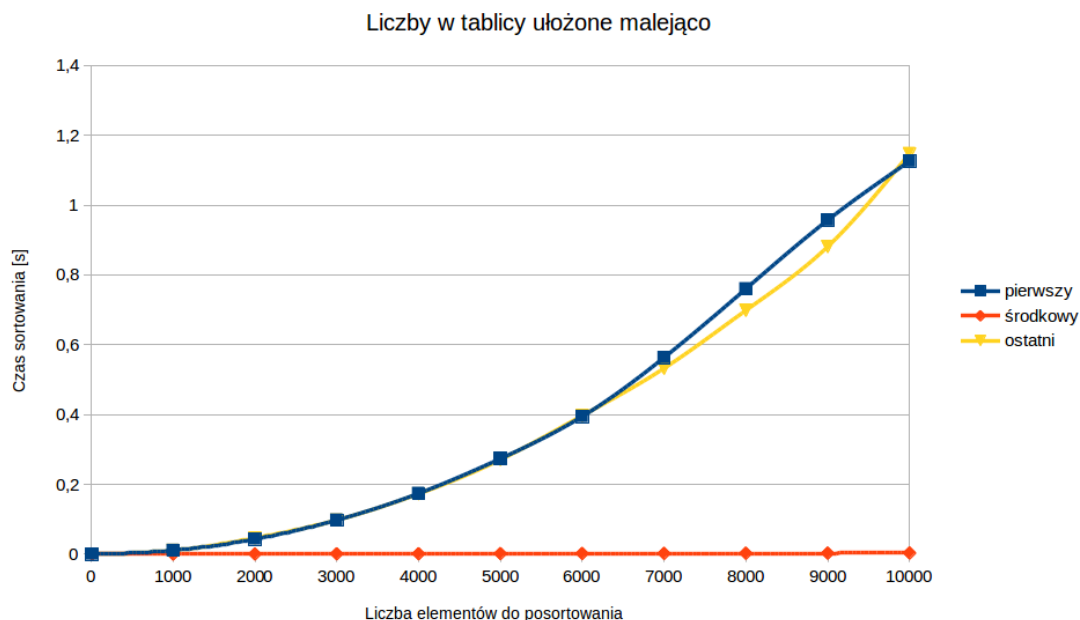


Na pierwszym wykresie liczby ułożone są losowo, jak widać czasy sortowania w zależności od wyboru pivotu są bardzo podobne. Złożoność obliczeniowa algorytmu zależy od branych pivotów, a ponieważ może wypaść zarówno

najlepszy jak i najgorszy przypadek to złożoność zawiera się w przedziale:  
 $n * \log(n) - n^2$ .



W tym przypadku elementy w tablicy ułożone są rosnąco. Łatwo zauważyć, iż wybór pivota pierwszego i ostatniego nie jest już tak efektywny jak wybór pivota środkowego. Co ciekawe wybór elementu pierwszego, nie jest równoważny z wyborem elementu ostatniego. Złożoność obliczeniowa sortowania z wyborem ostatniego elementu wynosi  $O(n) = n^2$



Na ostatnim wykresie elementy rozmieszczone są rosnąco. Złożoność obliczeniowa algorytmu sortowania przy wyborze pierwszego i ostatniego elementu jest podobna. Wybór elementu środkowego nadal jest najefektywniejszym rozwiązaniem.

### 3 Wnioski

Z przeprowadzonych testów wynika, że najgorszym sposobem sortowania szybkiego jest wybór skrajnego elementu tablicy. W najgorszym przypadku złożoność wynosi  $O(n) = n^2$ . Najlepszy możliwy przypadek to wybór mediany zbioru liczb, wtedy dostajemy złożoność  $O(n) = n \log n$ . Jednak jeżeli chcielibyśmy znaleźć taką medianę, wykonujemy dodatkowe przeszukiwanie tablicy, co znacznie pogarsza szybkość działania algorytmu.

Można wyróżnić również inne rozwiązania wyboru pivotu. Ciekawym rozwiązaniem wydaje się wybieranie losowego elementu zbioru, oczywiście istnieje prawdopodobieństwo na trafienie na najgorszy możliwy przypadek, jednak jest ono znacznie zredukowane - złożoność obliczeniowa około:  $O(n) = 1.386 * n \log(n)$ . Podobnie jest w przypadku wyboru środkowego elementu zbioru. Kolejnym rozwiązaniem jest wybranie mediany (pierwszego, środko-

wego oraz ostatniego elementu) jako pivota. Ten sposób rekomenduje Robert Sedgewick. Złożoność obliczeniowa wyniesie w tym przypadku około  $O(n) = 1.188 * n \log(n)$ .

## Arkusz3

	Pseudolosowe	Rosnąco	Malejąco
	pivot0	pivot0	pivot0
10	0,0000021	0,0000015	0,0000025
1000	0,0004281	0,0037623	0,0108436
2000	0,0010220	0,0210716	0,0433409
3000	0,0013207	0,0384276	0,0976138
4000	0,0016056	0,0595628	0,1740544
5000	0,0020619	0,0936939	0,2737472
6000	0,0025717	0,1374840	0,3945261
7000	0,0031947	0,1824460	0,5633722
8000	0,0035107	0,2405676	0,7606013
9000	0,0039629	0,3032257	0,9574657
10000	0,0044832	0,3769831	1,1263924
	pivot1	pivot1	pivot1
10	0,0000016	0,0000012	0,0000015
1000	0,0003320	0,0002193	0,0002488
2000	0,0007280	0,0004791	0,0005457
3000	0,0011646	0,0007774	0,0008826
4000	0,0015913	0,0010755	0,0012336
5000	0,0020917	0,0013534	0,0015861
6000	0,0025399	0,0017056	0,0019396
7000	0,0030793	0,0020051	0,0022141
8000	0,0035333	0,0022827	0,0025379
9000	0,0040425	0,0026110	0,0029038
10000	0,0044697	0,0029553	0,0033283
	pivot2	pivot2	pivot2
10	0,0000017	0,0000034	0,0000020
1000	0,0003311	0,0179559	0,0109396
2000	0,0007426	0,0726859	0,0462042
3000	0,0011774	0,1655679	0,0976339
4000	0,0016003	0,2871630	0,1727580
5000	0,0021162	0,4535743	0,2709232
6000	0,0026145	0,6457303	0,3969540
7000	0,0031589	0,8812976	0,5325851
8000	0,0035463	1,1516833	0,6987165
9000	0,0040277	1,4696148	0,8815530
10000	0,0044912	1,8050796	1,1483501