

Krawiec Andrzej
Sprawozdanie nr.2

Projekt nr.2 – Zapisanie i zbadanie algorytmów sortowania gnoma i sortowania kubełkowego

Opis zagadnień

W liście znajduję się x liczb z zakresu (0,n) – taki zakres przyjąłem przy badanych próbkach, badania przeprowadziłem kilkakrotnie na różnych zestawach danych przy użyciu algorytmów sortowania gnoma oraz sortowania kubełkowego, wcześniej zapoznałem się z działaniem tych dwóch algorytmów przy użyciu kilku stron internetowych

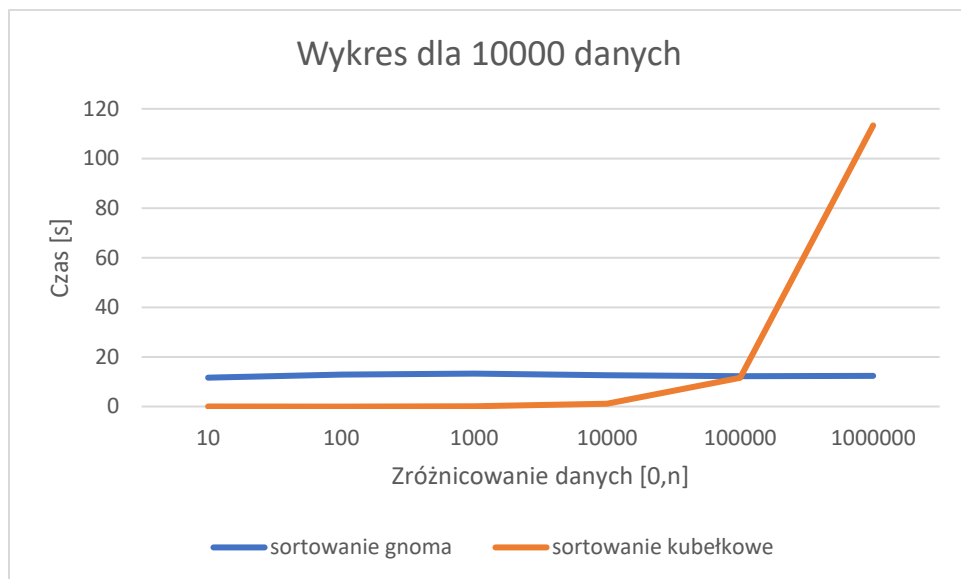
Dokumentacja z doświadczeń

10000 danych – sortowanie gnoma				
Zakres losowanych danych [0,n]	Czas – 1 próba [s]	Czas – 2 próba [s]	Czas – 3 próba [s]	Wynik uśredniony [s]
10	11,754335880279541	11.598498582839966	11.549288272857666	11,63404
100	13,045547246932983	12.849929332733154	12.586480855941772	12,82732
1000	13.236718654632568	13.415836095809937	13.146071910858154	13,26621
10000	12.872487306594849	12.617796897888184	12.40285587310791	12,63105
100000	12.027031660079956	12.154013395309448	12.47593379020691	12,21899
1000000	12.095223426818848	12.36277437210083	12.41250491142273	12,29017

Tabela 1. Sortowanie gnoma - 10000 danych

10000 danych – sortowanie kubełkowe				
Zakres losowanych danych [0,n]	Czas – 1 próba [s]	Czas – 2 próba [s]	Czas – 3 próba [s]	Wynik uśredniony [s]
10	0.01399540901184082	0.014979124069213867	0.015838146209716797	0,014938
100	0.015981435775756836	0.015929460525512695	0.01595902442932129	0,015957
1000	0.12589764595031738	0.12485527992248535	0.12725400924682617	0,126002
10000	1.1665918827056885	1.1821863651275635	1.1859827041625977	1,178254
100000	11.530770063400269	11.686786651611328	11.520776510238647	11,57944
1000000	114.67837953567505	112.87886619567871	112.3816192150116	113,313

Tabela 2. Sortowanie kubełkowe - 10000 danych



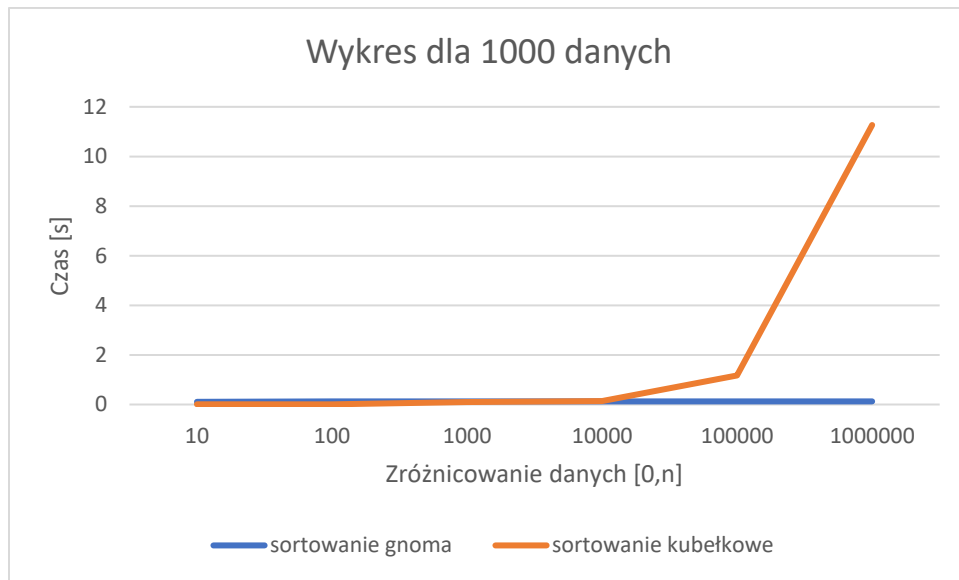
Rysunek 1. Porównanie dwóch sortowań - 10000 danych

1000 danych – sortowanie gnoma				
Zakres losowanych danych [0,n]	Czas – 1 próba [s]	Czas – 2 próba [s]	Czas – 3 próba [s]	Wynik uśredniony [s]
10	0.10272502899169922	0.11469149589538574	0.10973429679870605	0,10905
100	0.12166285514831543	0.11967968940734863	0.1311502456665039	0,124164
1000	0.12673068046569824	0.12734317779541016	0.12499237060546875	0,126355
10000	0.12569022178649902	0.12569022178649902	0.12569022178649902	0,12569
100000	0.12364459037780762	0.12364459037780762	0.12364459037780762	0,123645
1000000	0.12527012825012207	0.12617921829223633	0.12673139572143555	0,12606

Tabela 3. Sortowanie gnoma - 1000 danych

1000 danych – sortowanie kubełkowe				
Zakres losowanych danych [0,n]	Czas – 1 próba [s]	Czas – 2 próba [s]	Czas – 3 próba [s]	Wynik uśredniony [s]
10	0.012964725494384766	0.010963678359985352	0.009889602661132812	0,011273
100	0.0029916763305664062	0.008975744247436523	0.011967658996582031	0,007978
1000	0.015615701675415039	0.1436159610748291	0.12668800354003906	0,095307
10000	0.1414623260498047	0.12733197212219238	0.14144396781921387	0,136746
100000	1.186924695968628	1.169971227645874	1.1514520645141602	1,169449
1000000	11.310104608535767	11.273380994796753	11.217094898223877	11,26686

Tabela 4. Sortowanie kubełkowe - 1000 danych



Rysunek 2 . Porównanie dwóch sortowań - 1000 danych

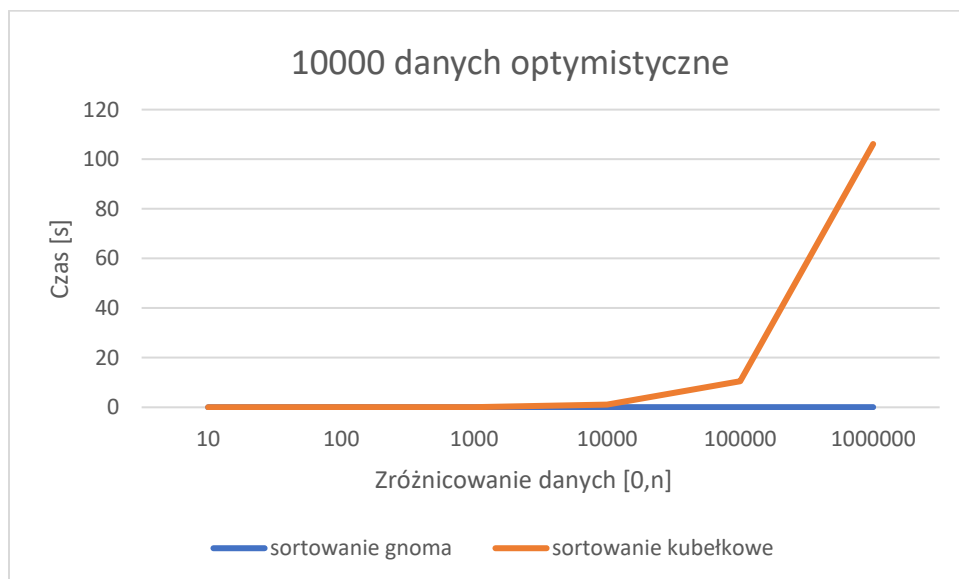
Kilka wykresów i porównań:

Optymistycznie – czyli lista jest już posortowana

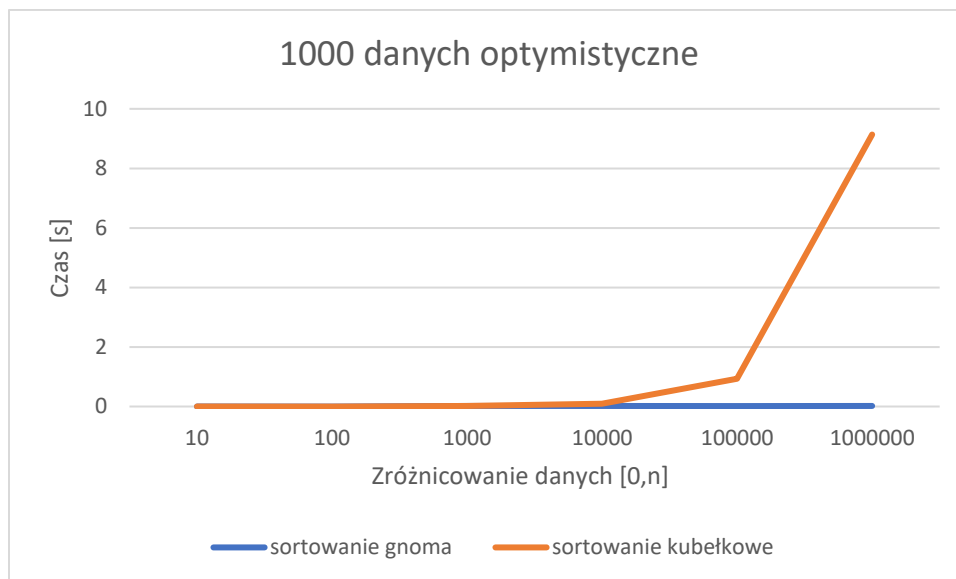
Pesymistycznie – lista jest posortowana ale odwrócona

Oczekiwane – losowe czyli liczby w liście są losowe w danym zakresie

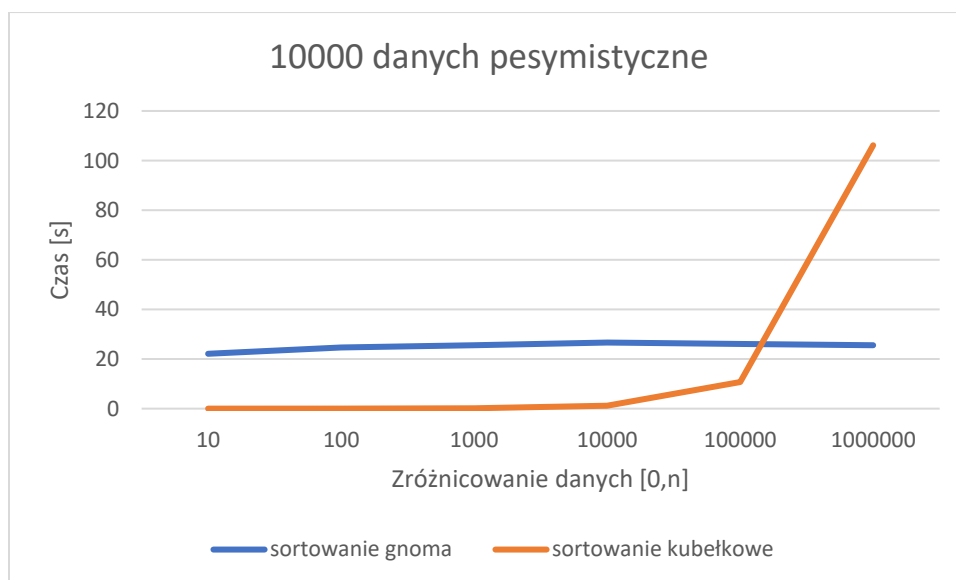
Wyniki były średnią z 3 prób.



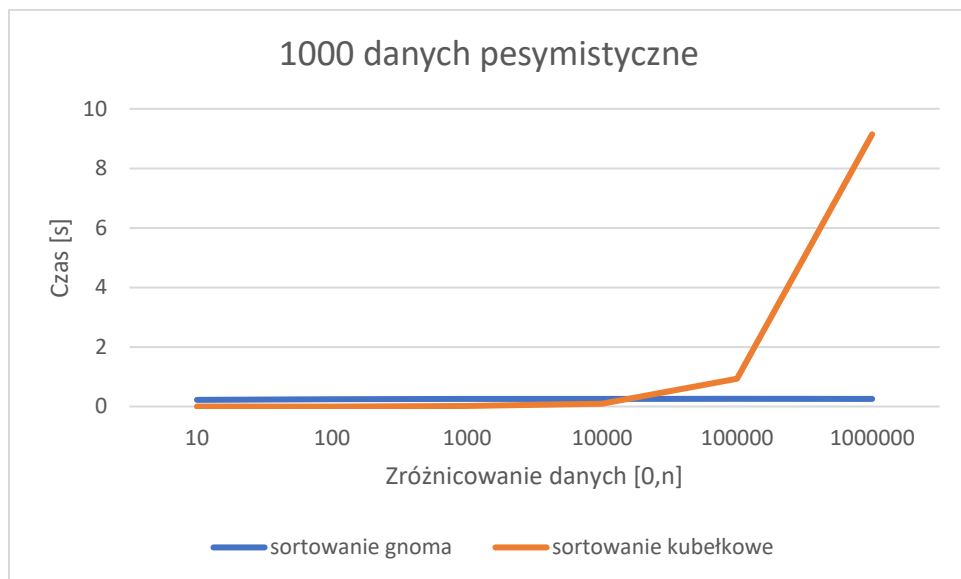
Rysunek 3. Sortowanie optymistyczne 10000 danych – lista jest już posortowana



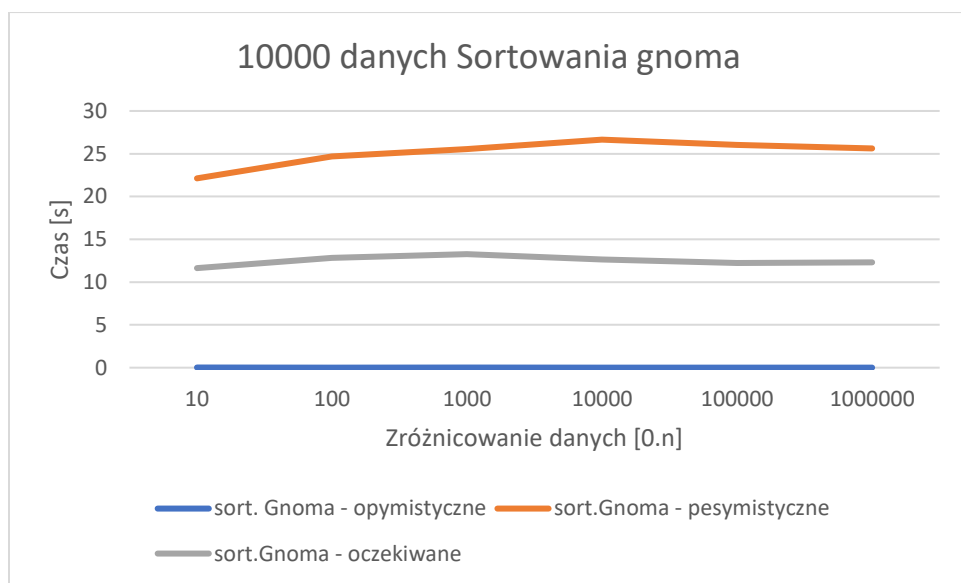
Rysunek 4. Sortowanie optymistyczne 1000 danych



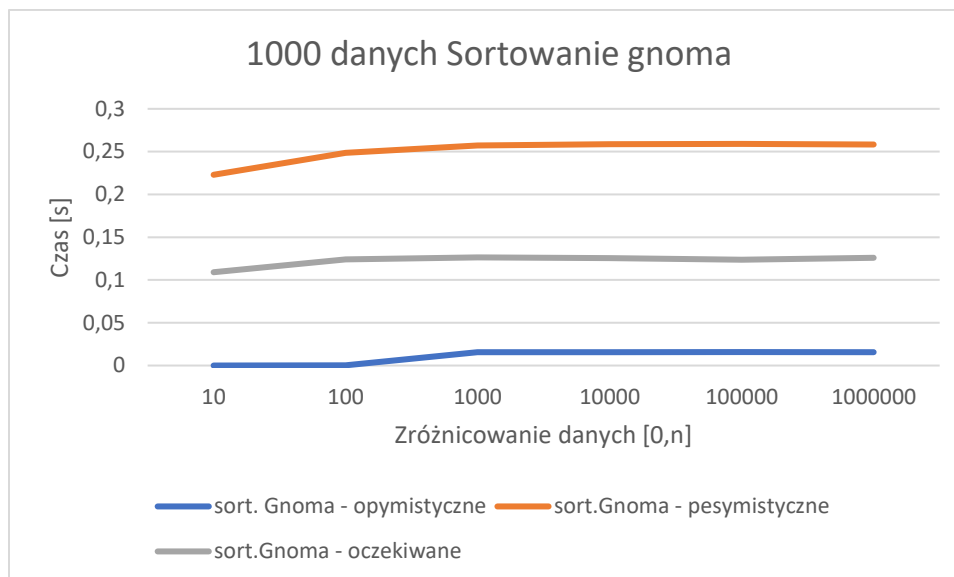
Rysunek 5. Sortowanie pesymistyczne 10000 danych



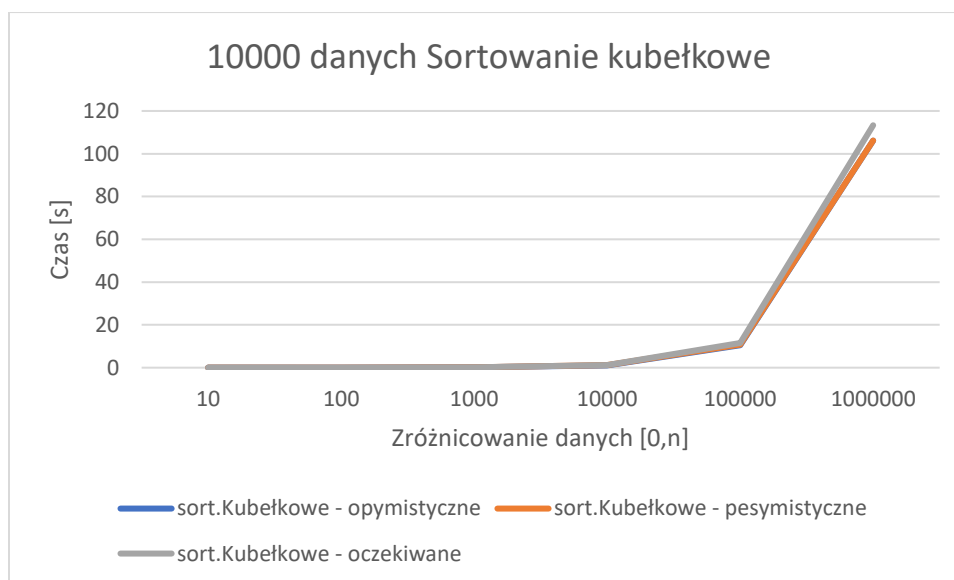
Rysunek 6. Sortowanie pesymistyczne 1000 danych – lista jest odwrócona



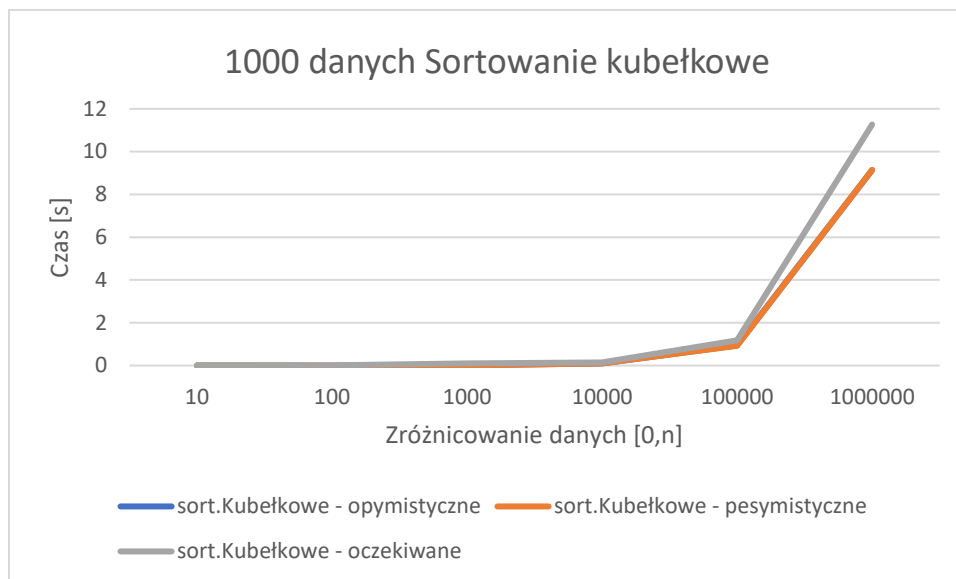
Rysunek 7. Różnice w sortowaniu gnoma zależne od danego przypadku - 10000 danych



Rysunek 8. Różnice w sortowaniu gnoma zależne od danego przypadku - 1000 danych



Rysunek 9. Różnice w sortowaniu kubełkowym zależne od danego przypadku - 10000 danych



Rysunek 10. Różnice w sortowaniu kubełkowym zależne od danego przypadku - 1000 danych

Jak możemy zauważyć na powyższych dwóch zdjęciach nie ma szczególnych różnic w sortowaniu kubełkowym ale to dlatego, że dane były losowane i były różne ilości wtedy kubełków. Dlatego dwa powyższe wykresy są bardziej poglądowymi wykresami odpowiadającymi na pytanie – jak radzi sobie sortowanie kubełkowe przy losowych danych dla 1000 i 10000 danych.

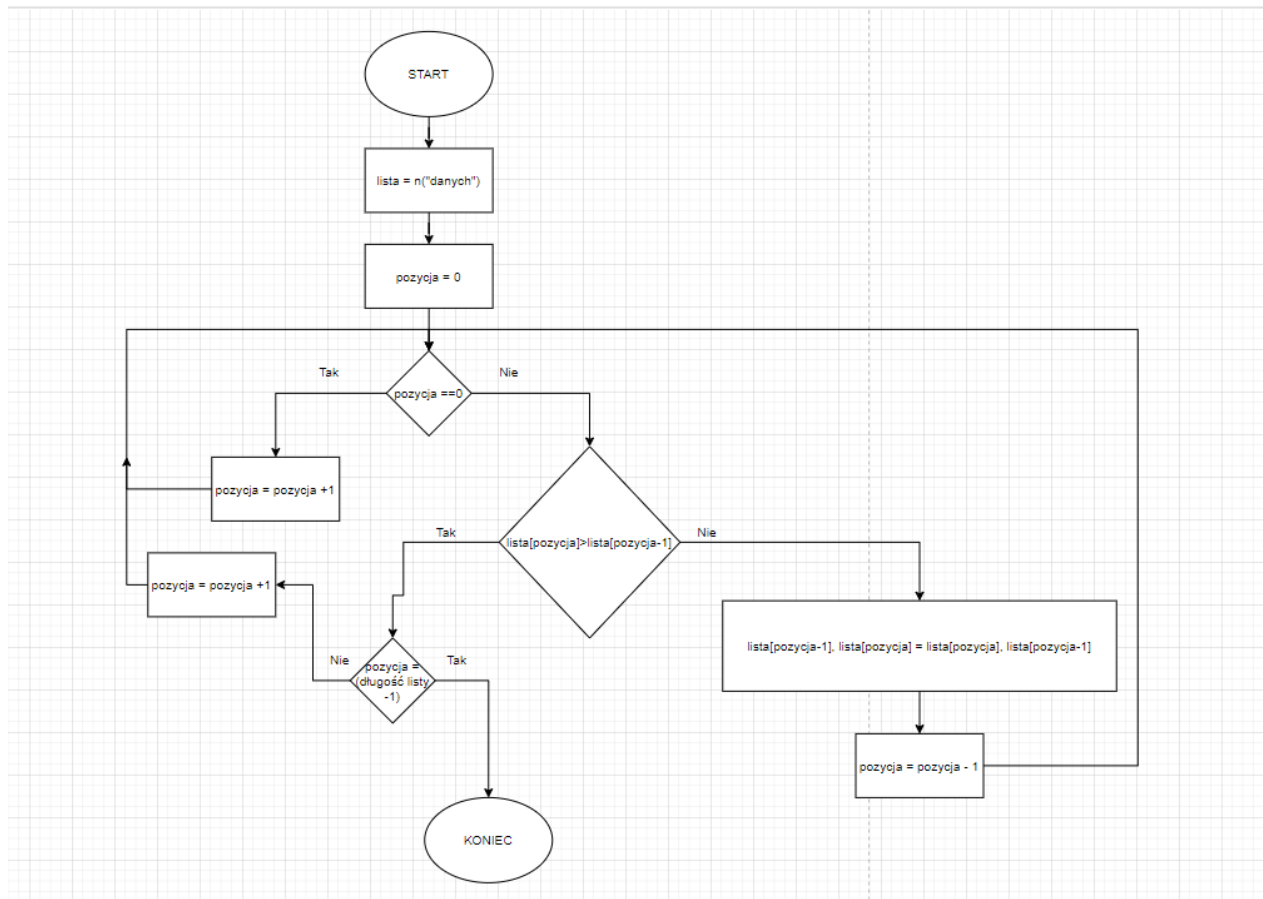
1000 danych – wyniki uśrednione z kilku prób	
Założenie optymistyczne – czas w [s]	Założenie pesymistyczne – czas w [s]
0	0,015892
10000 danych – wyniki uśrednione z kilku prób	
Założenie optymistyczne – czas w [s]	Założenie pesymistyczne – czas w [s]
0,015723	0,907232

Tabela 5. Sortowanie kubełkowe optymistyczne i pesymistyczne

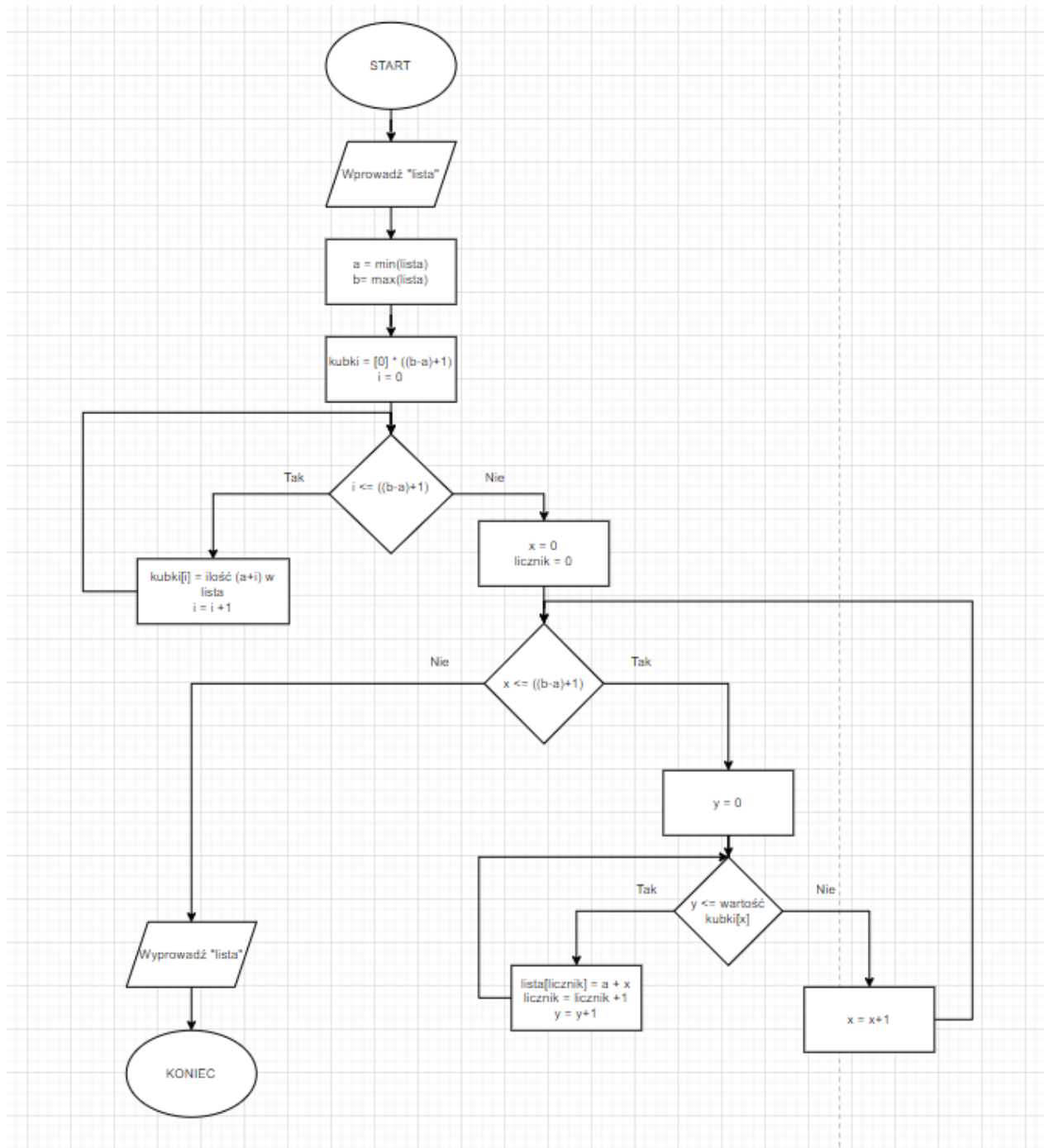
Gdzie zakładałem, że przykładem optymistycznym będzie lista w której będzie tylko jedna wartość, a pesymistyczny był gdy dla n liczby danych – w tym przypadku 1000 i 10000 będą w liście znajdować się wartości z całego tego przedziału bez powtórzeń przez co sortowanie kubełkowe stworzy n kubełków i każdy kubełek będzie zawierał jakąś wartość.

Schematy algorytmów:

Rysunek 11. Sortowanie gnoma - schemat



Rysunek 12. Sortowanie kubkowe - schemat



Pseudokod:

Sortowanie Gnomu:

1. Rozpoczęcie
2. Otworzenie pliku „plik.txt”
3. Pobieranie wartości z pliku do listy
4. Stworzenie dodatkowej zmiennej – licznik = 0

5. Dopóki licznik = długości listy wykonuj:
 - a. Jeżeli licznik = 0 to zwiększ o 1 jego wartość
 - i. Jeżeli lista[licznik] < lista[licznik – 1] to:
 1. Zamień te wartości miejscami
 2. Licznik zmniejsz o 1
6. Zapisanie posortowanej listy do pliku wyniki
7. Zamknięcie plików
8. Zakończenie programu.

Sortowanie kubełkowe:

1. Rozpoczęcie
2. Otworzenie pliku „plik.txt”
3. Pobieranie wartości z pliku do listy
4. Stworzenie dodatkowej zmiennej – $k = \text{maksymalny element w liście} - \text{minimalny element w liście} + 1$
5. Stworzenie kubełków(listy) i przypisanie im ilości wystąpień poszczególnych wartości zaczynając od elementu minimalnego w liście ale kubełki indeksujemy od 0 a następnie każdy kolejny kubełek ma szukać wartości o 1 większej od poprzedniego aż dojdzie to maksymalnego elementu
6. Odczytanie wartości z kubełków wiedząc że kubełek0 to ilość wystąpień elementu minimalnego w liście
7. Zapisanie posortowanej listy do pliku wyniki
8. Zamknięcie plików
9. Zakończenie programu.

Wnioski:

Jak jesteśmy w stanie zauważyć to sortowania które są pokazane wyżej mają swoje wady oraz zalety, jeśli chodzi o sortowanie kubełkowe to chciałbym zauważyć, że w przeprowadzonym badaniu nie dzieliłem danego przedziału liczbowego na mniejsze przedziały i do nich dopisywałem liczby, a wiem że są też takie techniki, w wykorzystanym sortowaniu kubełkowym przedział liczbowy każdy kubełek miał jedną wartość, a nie ich zakres.

Po wykonaniu doświadczenia oraz wykresów do jego wyników widzimy ciekawą zależność – otóż sortowanie gnoma zależy od ilości danych, a sortowanie kubełkowe zależy od zróżnicowania danych dzięki czemu jesteśmy w stanie dopasować odpowiednie sortowanie do zestawu danych, gdy potrzebujemy sortowania dużej ilości danych które są zróżnicowane to lepiej wybrać sortowanie gnoma ale gdy mamy dużą ilość danych ale ich rozrzut wartości jest stosunkowo niewielki to lepszym rozwiązaniem będzie zastosowanie sortowania kubełkowego.

Złożoności czasowa:

Dla sortowanie gnoma to:

- $O(n)$ dla przypadku optymistycznego gdy lista jest już posortowana
- $O(n^2)$ dla przypadku pesymistycznego gdy lista jest posortowana w „drugą stronę”

Dla sortowania kubełkowego to:

- $O(n)$ dla przypadku optymistycznego
- $O(n^2)$ dla przypadku najbardziej pesymistycznego

Spis tabel:

Tabela 1. Sortowanie gnoma - 10000 danych	1
Tabela 2. Sortowanie kubełkowe - 10000 danych	1
Tabela 3. Sortowanie gnoma - 1000 danych	2
Tabela 4. Sortowanie kubełkowe - 1000 danych	2
Tabela 5. Sortowanie kubełkowe optymistyczne i pesymistyczne.....	7

Spis rysunków:

Rysunek 1. Porównanie dwóch sortowań - 10000 danych	2
Rysunek 2 . Porównanie dwóch sortowań - 1000 danych	3
Rysunek 3. Sortowanie optymistyczne 10000 danych	3
Rysunek 4. Sortowanie optymistyczne 1000 danych	4
Rysunek 5. Sortowanie pesymistyczne 10000 danych	4
Rysunek 6. Sortowanie pesymistyczne 1000 danych	5
Rysunek 7. Różnice w sortowaniu gnoma zależne od danego przypadku - 10000 danych	5
Rysunek 8. Różnice w sortowaniu gnoma zależne od danego przypadku - 1000 danych	6
Rysunek 9. Różnice w sortowaniu kubełkowym zależne od danego przypadku - 10000 danych	6
Rysunek 10. Różnice w sortowaniu kubełkowym zależne od danego przypadku - 1000 danych	7

Strony z jakich korzystałem w celu nauki i poznania algorytmów:

- <https://stackoverflow.com/questions/13167300/python-simple-swap-function>
- <https://mattomatti.com/pl/fs10>
- https://pl.wikipedia.org/wiki/Sortowanie_gnoma
- https://pl.wikipedia.org/wiki/Sortowanie_kube%C5%82kowe
- <https://binarnie.pl/sortowanie-kubełkowe/>