// Autor: Daniel Szarek
//====================

Zawartość folderu z zadaniem:
main.cpp – Kod z rozwiązaniem zadania napisanym w języku C++
funkcje.h – Biblioteka z własnymi funkcjami wykorzystywanymi do rozwiązania zadania N9
opracowanie.pdf – Ten dokument z opracowaniem zadania N9
funkcje.cpp – Plik zawierający implementację funkcji podanych w własnej bibliotece funkcje.h
Makefile – Do uruchomienia programu main.cpp (Makefile oferuje możliwość uruchomienia programu za pomocą komendy 'make run', usunięcia plików po uruchomieniu programu komendą 'make clean' oraz możliwość utworzenia paczki .tar.gz z zawartością foldera z zadaniem za pomocą komendy 'make tar'
wyniki1.txt – Plik tekstowy z wektorem własnym dla pierwszej wartości własnej macierzy A
wyniki2.txt – Plik tekstowy z wektorem własnym dla drugiej wartości własnej macierzy A
wyniki3.txt – Plik tekstowy z wektorem własnym dla trzeciej wartości własnej macierzy A
wyniki4.txt – Plik tekstowy z wektorem własnym dla czwartej wartości własnej macierzy A
folder: wykresy zawiera wykresy do graficznej wizualizacji danych wektorów własnych, wykres ogólny z wszystkimi czterema wektorami naraz oraz wykresy porównujący wszystkie wartości wektorów własnych na jednym wykresie.
folder: skrypty gnuplot zawierający skrypty do programu gnuplot do utworzenia wyżej wspomnianych wykresów

Do zrealizowania zadania wykorzystałem GSL - GNU Scientific Library, w Makefile zawarłem komendy do uruchomienia zadania na swojej maszynie w systemie Ubuntu, w celu uruchomienia zadania na swoim sprzęcie należy w prawidłowy sposób zainstalować bibliotekę GSL oraz zmienić scieżkę INCLUDEPATH1 dla swojej maszyny.

W swoich plikach źródłowych zamieszczam liczne komentarze, które na bieżąco tłumaczą działanie oraz funkcjonalności mojego kodu. W tym dokumencie przedstawię omówienie metod wykorzystanych do rozwiązania zadania.

Treść Zadania:

N9 *Zadanie numeryczne*
Znaleźć cztery najmniejsze wartości i wektory własne macierzy $N \times N$ z dokładnością 10^{-8}, której elementy są dane jako

$$A_{nm} = -\frac{\delta_{n-1,m}}{h^2} + \left(\frac{2}{h^2} + 4 - \frac{6}{\cosh^2 x_n}\right)\delta_{n,m} - \frac{\delta_{n+1,m}}{h^2}\,, \tag{10}$$

gdzie N duże $\sim 100 \ldots 1000$,

$$h = \frac{2L}{N-1} \qquad x_n = -L + nh\,, \qquad L = 10. \tag{11}$$

Cztery znalezione wektory proszę przedstawić w postaci graficznej - wykresu (x_n, y_n), gdzie y_n $n-$ta składowa wektora.

Rozwiązanie zadania:

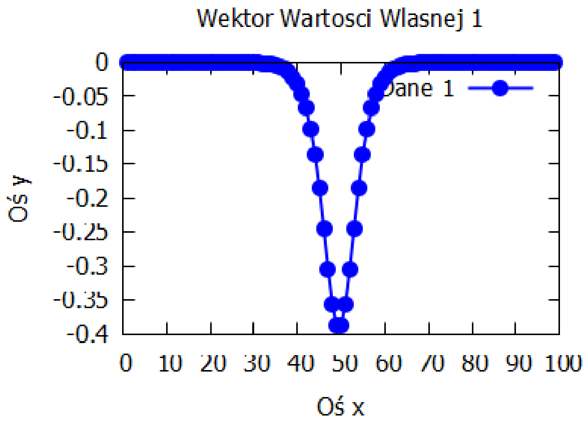
```
vboxuser@Ubuntu:~/Downloads/Zadanie9$ make run
g++ -Wall -std=c++11 -I/home/vboxuser/Desktop/gsl/include -c main.cpp -o main.o
g++ -Wall -std=c++11 -I/home/vboxuser/Desktop/gsl/include -c funkcje.cpp -o funkcje.o
ar rsv libMojeFunkcje.a funkcje.o
ar: creating libMojeFunkcje.a
a - funkcje.o
mkdir -p ./lib
mv libMojeFunkcje.a ./lib
g++ -o main.x main.o -Wall -std=c++11 -L./lib -lMojeFunkcje -lgsl -lgslcblas
./main.x
Wartosc wlasna macierzy A:
Î»[0] = -0.0078467168
Wektor wlasny macierzy A:
-1.92344e-09
-4.16149e-09
-7.08024e-09
-1.11571e-08
-1.70589e-08
-2.5751e-08
-3.86552e-08
-5.78821e-08
-8.65768e-08
-1.29433e-07
-1.9346e-07
-2.89131e-07
-4.32095e-07
-6.45736e-07
-9.64999e-07
```

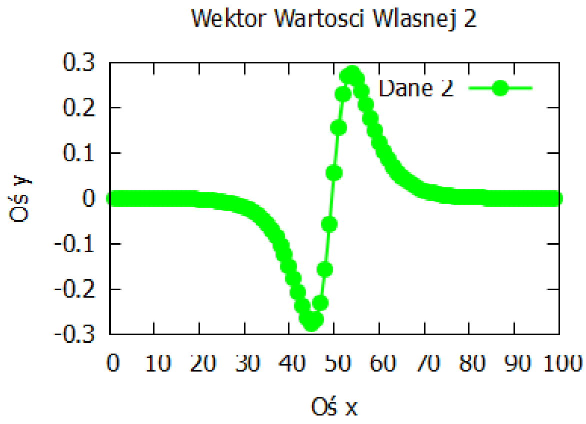
```
Wartosc wlasna macierzy A:
Î»[1] = 2.9846521651
Wektor wlasny macierzy A:
-1.56722e-05
-3.19939e-05
-4.96414e-05
-6.93459e-05
-9.1924e-05
-0.000118311
-0.000149601
-0.00018709
-0.000232332
-0.000287202
-0.000353973
-0.000435411
-0.000534893
-0.000656539
-0.000805392
-0.000987618
-0.00121077
-0.00148409
-0.00181891
-0.00222909
-0.00273164
-0.00334736
-0.00410176
-0.00502607
-0.00615854
-0.00754602
-0.00924581
-0.0113281
```

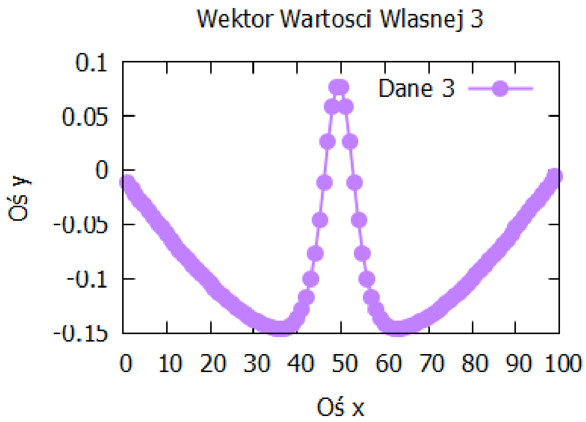
```
Wartosc wlasna macierzy A:
Î»[2] = 4.0295776013
Wektor wlasny macierzy A:
-0.00540345
-0.0108004
-0.0161843
-0.0215486
-0.0268869
-0.0321928
-0.0374598
-0.0426816
-0.0478519
-0.0529644
-0.058013
-0.0629915
-0.067894
-0.0727145
-0.0774472
-0.0820864
-0.0866264
-0.0910618
-0.095387
-0.0995968
-0.103686
-0.107649
-0.111482
-0.115178
-0.118732
-0.12214
-0.125393
```

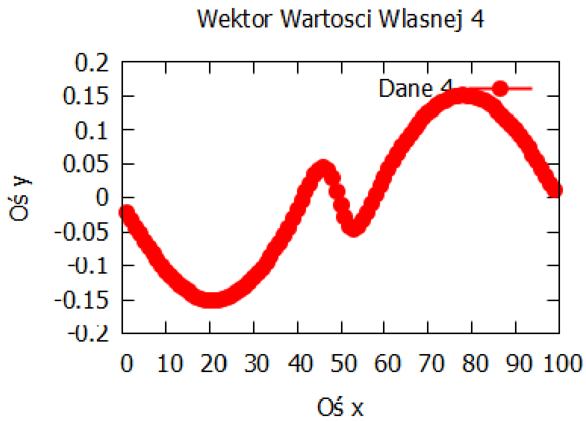
```
Wartosc wlasna macierzy A:
Î»[3] = 4.1280804242
Wektor wlasny macierzy A:
-0.0109113
-0.0217656
-0.0325061
-0.0430767
-0.0534221
-0.0634882
-0.0732225
-0.0825741
-0.091494
-0.0999356
-0.107855
-0.11521
-0.121963
-0.128079
-0.133525
-0.138273
-0.142298
-0.145579
-0.148099
-0.149845
-0.150806
-0.150978
-0.150359
-0.148953
-0.146764
-0.143803
-0.140084
-0.135622
-0.130436
-0.124547
-0.117976
-0.110744
-0.102872
```

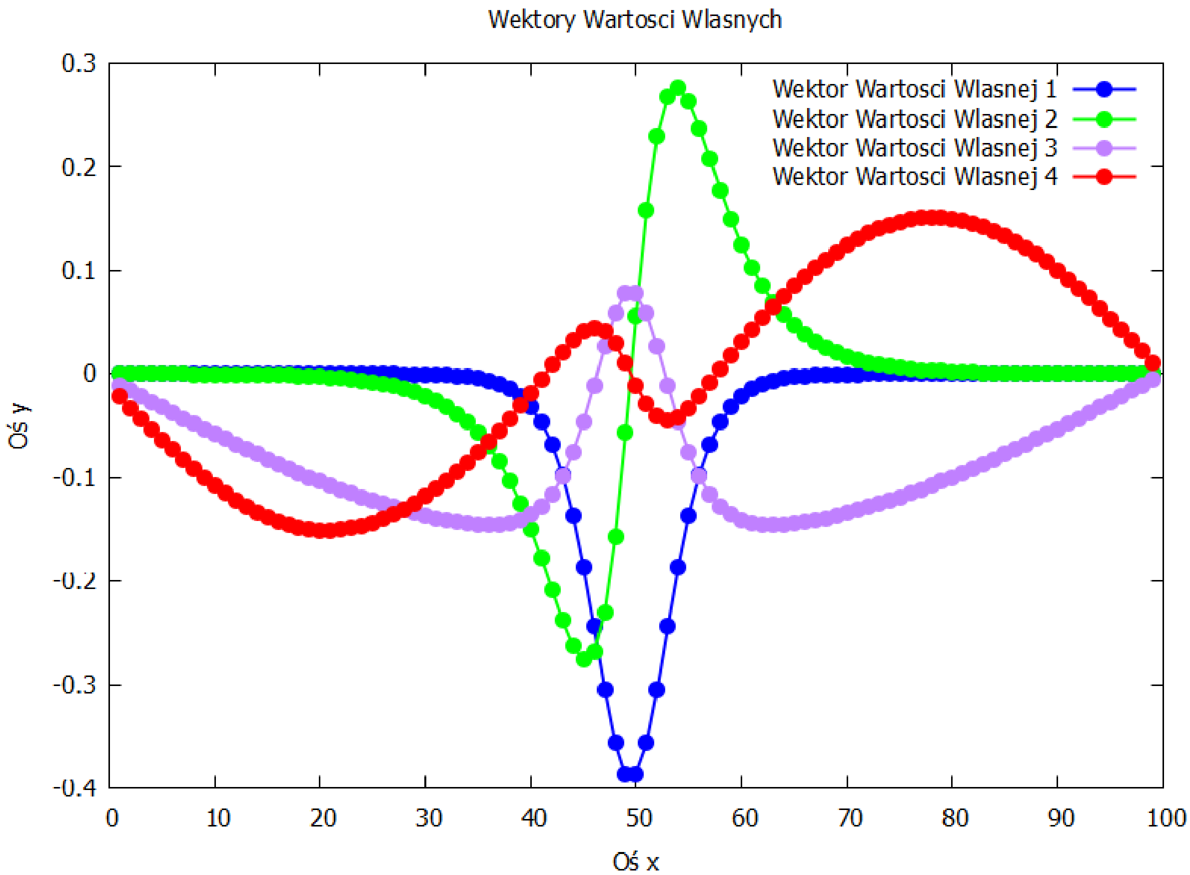
Poniżej wykresy ukazujące w sposób graficzny pełne wartości wektorów własnych:


Wektor Wartosci Wlasnej 1
Dane 1
Oś y
Oś x


Wektor Wartosci Wlasnej 2
Dane 2
Oś y
Oś x


Wektor Wartosci Wlasnej 3
Dane 3
Oś y
Oś x


Wektor Wartosci Wlasnej 4
Dane 4
Oś y
Oś x


Wektory Wartosci Wlasnych
Wektor Wartosci Wlasnej 1
Wektor Wartosci Wlasnej 2
Wektor Wartosci Wlasnej 3
Wektor Wartosci Wlasnej 4
Oś y
Oś x

Omówienie:

W ramach rozwiązania zadania z przedziału [100,1000], wybrałem jako N wartość 100.

W celu rozwiązania zadania należy zidentyfikować macierz A oraz dobrać do niej odpowiednią metodę znajdowania najmniejszych wartości własnych tej macierzy oraz odpowiadającym im wektorom własnym.

W pierwszej kolejności chciałbym się skupić na omówieniu macierzy A

Macierz A to macierz trójdiagonalna:

$$A = \begin{pmatrix} \frac{2}{h^2} + 4 - \frac{6}{\cosh^2 x_0}, & -\frac{1}{h^2}, & 0, & 0, & \dots \\ -\frac{1}{h^2}, & \frac{2}{h^2} + 4 - \frac{6}{\cosh^2 x_1}, & -\frac{1}{h^2}, & 0, & \dots \\ \vdots & & \ddots & & \end{pmatrix}$$

Zawartość wektora diagonalnego: $\left(\frac{2}{h^2} + 4 - \frac{6}{\cosh^2 x_n}\right)$

Zawartość wektora:

supdiagonalnego: $-\frac{\delta_{n-1,m}}{h^2}$

subdiagonalnego: $-\frac{\delta_{n+1,m}}{h^2}$

Będą one wypełnione tymi samymi wartościami

Jak widać macierz będzie specjalnym przypadkiem macierzy trójdiagonalnej o takich samych wartościach na supdiagonali oraz subdiagonali zwaną macierzom bidiagonalną główną.

Stąd w swoim programie dla oszczędności operuje na trzech wektorach macierzy A: diagonalnym i wcześniej wspomnianych supdiagonalnym oraz subdiagonalnym.

W celu znalezienia najmniejszych wartości własnych oraz odpowiadającym im wektorom własnym macierzy A wykorzystałem odwróconą metodę potęgową, która jest modyfikacją standardowej metody potęgowej omówionej w opracowaniu poprzedniego zadania.

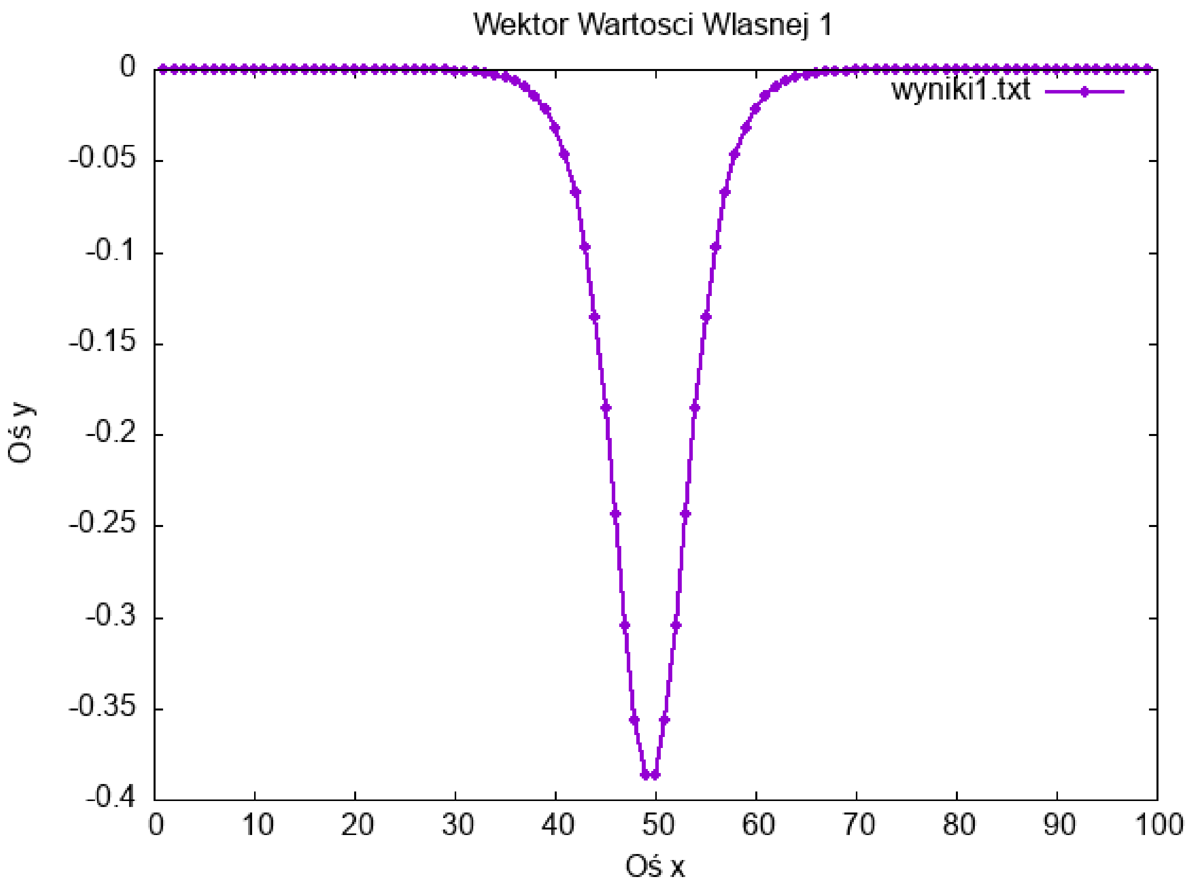
Do zalet tej metody należy fakt, że nie wymaga ona podania informacji startowych typu np. przesunięcie bliskie poszukiwanym wartościom własnym tak jak to ma miejsce w odwróconej metodzie potęgowej z przesunięciem oraz to, że ta metoda jest stabilna. To nie oznacza, że ta metoda jest pozbawiona wad, ponieważ brak podania informacji kosztuję czas wykonania programu, ogólnie powinna mieć ona więcej iteracji niż wcześniej wspomniany jej odpowiednik z przesunięciem. W dodatku metoda ta jest wrażliwa na dobór wektora początkowego. Mimo wszystko sposób ten jest najbardziej uniwersalny i pozwala na optymalne rozwiązanie tego zadania.

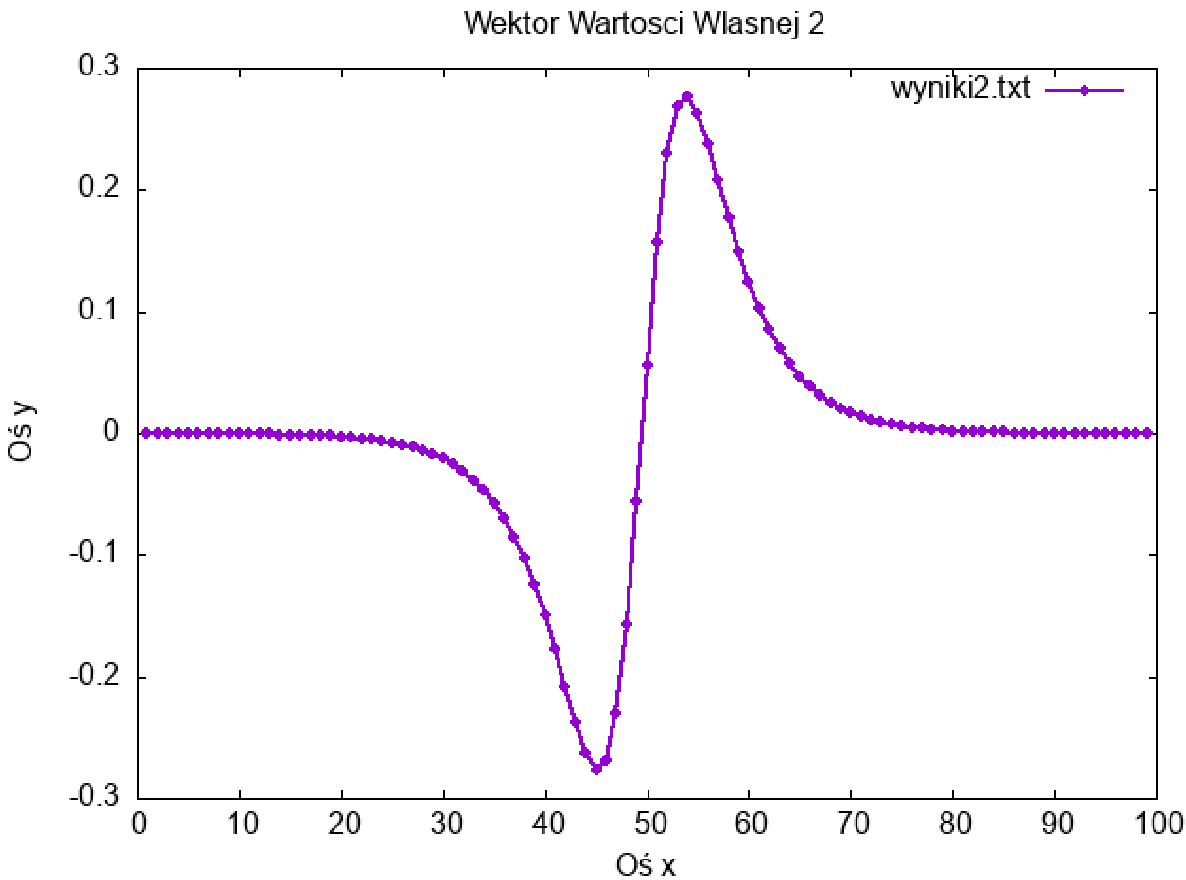
Poniżej wklejam algorytm ze strony https://mathreview.uwaterloo.ca/archive/voli/1/panju.pdf / na którym bazowałem swoją implementację w kodzie. Strona ta została podana w treści poprzedniego zadania N8.

Pick a starting vector $v^{(0)}$ with $\left\|v^{(0)}\right\| = 1$
For $k = 1, 2, \dots$
    Solve $Aw = v^{(k-1)}$ for w
    Let $v^{(k)} = w / \|w\|$

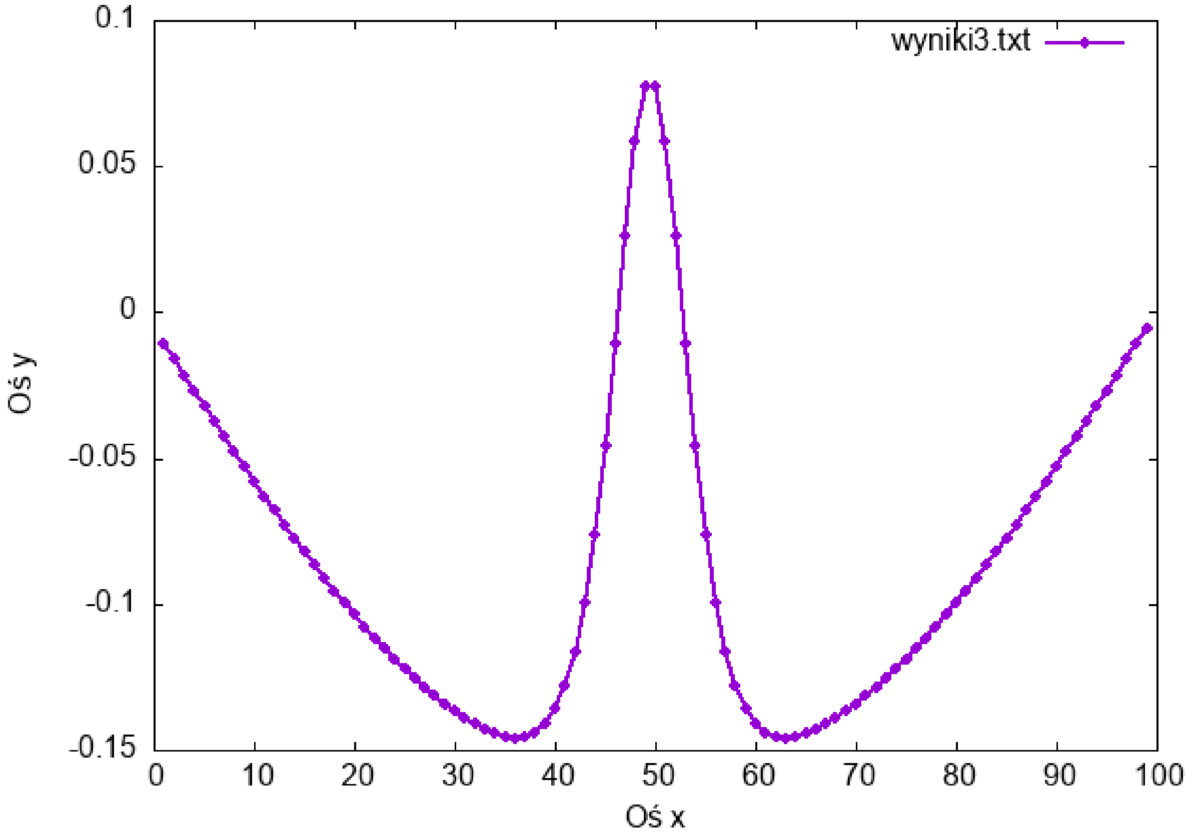
Tak jak to wspomniałem w opracowaniu poprzedniego zadania N8, tutaj wykorzystam dodatkowo metodę Rayleigha, celu uzyskania szybszej zbieżności. Wykorzystanie tej metody ma miejsce po uzyskaniu w miarę dobrego przybliżenia najmniejszych wartości własnych Operacja ta jest powielana dla kolejnych wartości własnych, uzyskiwanym dzięki wykorzystaniu deflacji wektorowej.

Poniżej zamieszczam osobne wykresy wektorów własnych dla najmniejszych wartości własnych macierzy A:


Wektor Wartosci Wlasnej 1
wyniki1.txt
0
-0.05
-0.1
-0.15
-0.2
-0.25
-0.3
-0.35
-0.4
Oś y
0
10
20
30
40
50
60
70
80
90
100
Oś x


Wektor Wartosci Wlasnej 2
wyniki2.txt
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
Oś y
0
10
20
30
40
50
60
70
80
90
100
Oś x

Wektor Wartosci Wlasnej 3



Wektor Wartosci Wlasnej 4