

## Raport z laboratoriów 6 – układy równań liniowych

### → Uwagi wstępne:

Program na początku po jego uruchomieniu pyta, które ćwiczenie (który case w switch'u) ma wykonać. Należy podać numer ćwiczenia. Program korzysta z dołączonego pliku gauss.cpp, w którym należy na samej górze zdefiniować dodatkowo N, tj napisać `const int N=50`, żeby program działał poprawnie (okazało się, że nie wystarczy zadeklarować stałej w pliku main.cpp; trzeba tę czynność również powtórzyć w gauss.cpp).

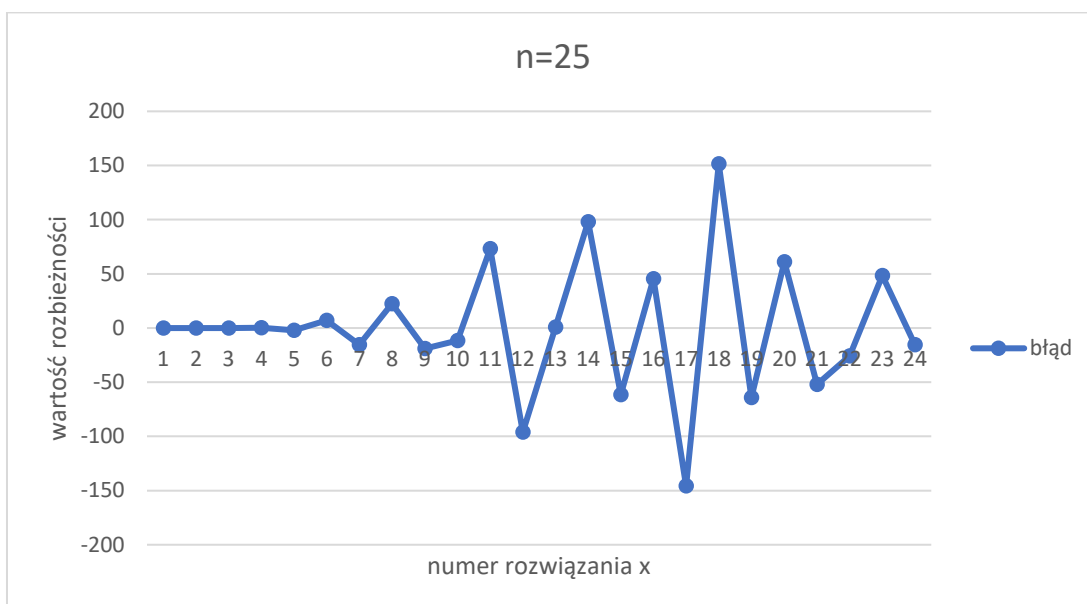
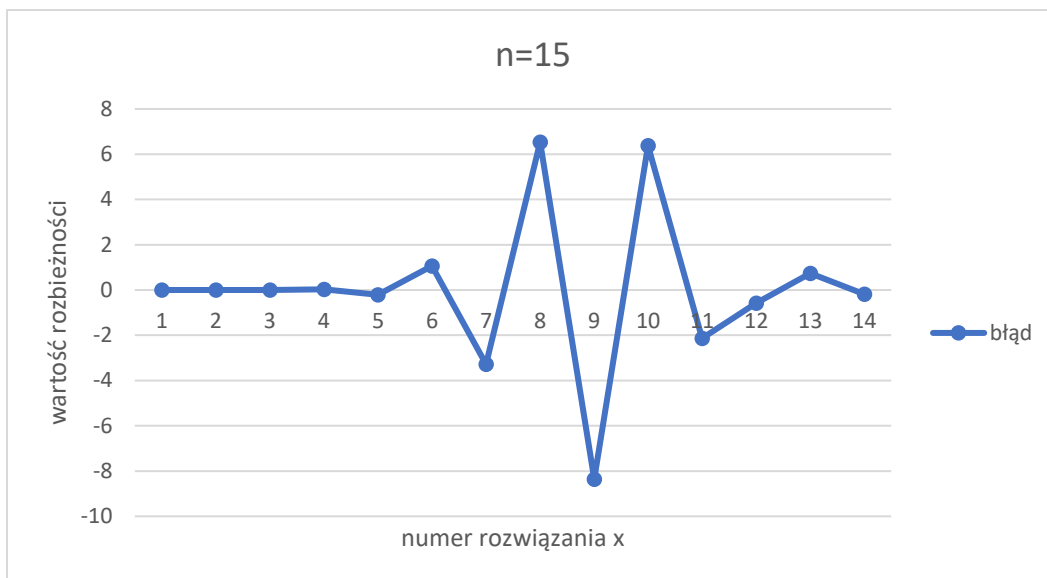
### → Ćwiczenie 1 (w kodzie case:1)

Celem tej części kodu jest rozwiązanie układu równań liniowych metodą Gaussa, gdzie macierz współczynników jest macierzą Hilberta a macierz wyrazów wolnych jest sumą współczynników w danym wierszu. Napisałem funkcje, o których jest mowa w zadaniu 2 w instrukcji, rozwiązałem układ równań z wykorzystaniem metody Gaussa dla różnych n (dla różnych wymiarów macierzy  $n \times n$ ), oto kilka wyników rozwiązań układu:

<pre> Wyniki dla n=4 1.000000 1.000000 1.000000 1.000000 </pre>	<pre> Wyniki dla n=10 1.000000 1.000000 0.999998 1.000020 0.999903 1.000267 0.999563 1.000421 0.999779 1.000048 </pre>	<pre> Wyniki dla n=20 1.000000 1.000075 0.997232 1.043007 0.657461 2.485524 -2.218021 2.012694 11.914026 -20.728838 5.732215 34.233277 -42.153623 18.727738 2.721547 -16.803379 49.405743 -57.516277 33.288344 -5.798746 </pre>
---	--	---

Zgodnie z instrukcją wartość parametru n należy zmieniać ręcznie w kodzie. Domyślna wartość n to 6, czyli program liczy rozwiązanie metodą Gaussa dla sześciu równań z sześcioma niewiadomymi.

A tutaj wykresy rozbieżności wyników dla kolejnych rozwiązań dla konkretnych n:



### ➔ Ćwiczenie 2 (w kodzie case:2)

Celem tej części kodu jest wyznaczenie macierzy trójkątnej. Ta część programu domyślnie jest realizowana dla  $n=7$ , jednakże to można zmienić ręcznie w kodzie. W kodzie znajdują się również szczegółowe komentarze po kolei co jest liczone, zatem tutaj umieszczę jedynie przykładowe wyniki:

```

F:\WOJTEK\STUDIA\SEM2\informatyka 2\laby6\Debug\laby6.exe
1 - rozwiazanie metoda Gaussa
2 - obliczane macierzy trojkatnej oraz wyznacznika
3 -

Podaj numer cwiczenia: 2

Wyniki dla n = 7
Macierz wyjsciowa
1.000000 0.500000 0.333333 0.250000 0.200000 0.166667 0.142857
0.500000 0.333333 0.250000 0.200000 0.166667 0.142857 0.125000
0.333333 0.250000 0.200000 0.166667 0.142857 0.125000 0.111111
0.250000 0.200000 0.166667 0.142857 0.125000 0.111111 0.100000
0.200000 0.166667 0.142857 0.125000 0.111111 0.100000 0.090909
0.166667 0.142857 0.125000 0.111111 0.100000 0.090909 0.083333
0.142857 0.125000 0.111111 0.100000 0.090909 0.083333 0.076923

Macierz zdiagonalizowana
1.000000 0.500000 0.333333 0.250000 0.200000 0.166667 0.142857
0.000000 0.083333 0.083333 0.075000 0.066667 0.059524 0.053571
0.000000 -0.000000 0.005556 0.008333 0.009524 0.009921 0.009921
0.000000 0.000000 0.000000 0.000357 0.000714 0.000992 0.001190
0.000000 0.000000 0.000000 0.000000 0.000023 0.000057 0.000093
0.000000 0.000000 0.000000 0.000000 0.000000 0.000001 0.000004
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000

wyznacznik = 4.8358e-25_

```

```

F:\WOJTEK\STUDIA\SEM2\informatyka 2\laby6\Debug\laby6.exe
1 - rozwiazanie metoda Gaussa
2 - obliczane macierzy trojkatnej oraz wyznacznika
3 -

Podaj numer cwiczenia: 2

Wyniki dla n = 4
Macierz wyjsciowa
1.000000 0.500000 0.333333 0.250000
0.500000 0.333333 0.250000 0.200000
0.333333 0.250000 0.200000 0.166667
0.250000 0.200000 0.166667 0.142857

Macierz zdiagonalizowana
1.000000 0.500000 0.333333 0.250000
0.000000 0.083333 0.083333 0.075000
0.000000 -0.000000 0.005556 0.008333
0.000000 0.000000 0.000000 0.000357

wyznacznik = 1.65344e-07

```

### ➔ Ćwiczenie 3: (w kodzie case:3)

Program ma czytać macierz oraz wektor b z wcześniej utworzonych plików. Żeby program był bardziej uniwersalny w pliku z danymi pierwsza wartość musi określać liczbę równań w układzie. Dalej plik ma zawierać wypisaną macierz A, a pod jej ostatnim elementem ma się znajdować wektor b. Program będzie po kolei czytał wartości dla odpowiednio kolejnych wierszy i kolumn, następnie wczyta wektor b i obliczy x za pomocą metody Gaussa. Do tego celu muszę zastosować dynamiczną alokację pamięci. W dodatku potrzebuję zmodyfikować funkcje wyświetlające macierz czy wektor. W tym celu napisałem nowe funkcje: *DisplayMatrixw(n, M)* oraz *DisplayVecw(n, b)*, które jako argumenty przyjmują odpowiednio wskaźnik do macierzy i wektora (zmiana jest spowodowana dynamicznie zaalokowanym wektorem i macierzą). Ponadto napisałem nową funkcję *void gaussw(int n, double \*\*a, double \*b, double \*x)*, która różni się od bazowej funkcji zawartej w pliku *gauss.cpp* jedynie rodzajem przyjmowanych argumentów.

- Program wczytuje wartości w sposób wyżej omówiony z pliku2.txt, w którym znajduje się „macierz Hilberta”, następnie wczytuje wektor

```
plik2 — Notatnik
Plik Edycja Format Widok Pomoc
6
1. 0.000000      0.500000      0.333333      0.250000      0.200000      0.166667
0.500000      0.333333      0.250000      0.200000      0.166667      0.142857
0.333333      0.250000      0.200000      0.166667      0.142857      0.125000
0.250000      0.200000      0.166667      0.142857      0.125000      0.111111
0.200000      0.166667      0.142857      0.125000      0.111111      0.100000
0.166667      0.142857      0.125000      0.111111      0.100000      0.090909

2. 4.50000
1.592857
1.217857
0.995635
0.845635
0.736544
```

- Dla podanych wartości wylicza wektor  $x$

```
F:\WOJTEK\STUDIA\SEM2\informatyka 2\laby6\Debug\laby6.exe
1 - rozwiązanie metoda Gaussa
2 - obliczanie macierzy trojkatnej oraz wyznacznika
3 -

Podaj numer cwiczenia: 3
wczytana macierz:
1. 0.000000      0.500000      0.333333      0.250000      0.200000      0.166667
0.500000      0.333333      0.250000      0.200000      0.166667      0.142857
0.333333      0.250000      0.200000      0.166667      0.142857      0.125000
0.250000      0.200000      0.166667      0.142857      0.125000      0.111111
0.200000      0.166667      0.142857      0.125000      0.111111      0.100000
0.166667      0.142857      0.125000      0.111111      0.100000      0.090909

wczytany wektor prawych stron:
2.450000
1.592857
1.217857
0.995635
0.845635
0.736544

Wyniki
1. 0.000000
1. 0.000000
1. 0.000000
1. 0.000000
1. 0.000000
1. 0.000000
1. 0.000000
-
```

Wyniki nie są zaskakujące; wczytałem macierz Hilberta i wektor wyrazów wolnych jej odpowiadający i otrzymałem wyniki jak poprzednio. Sugeruje to, że program działa poprawnie.

#### ➔ Ćwiczenie 4: (w kodzie case 4:)

Celem tej części kodu jest obliczenie metodą Gaussa rozwiązań układu równań wykorzystując dynamiczną alokację pamięci. Jest to niejako połączenie ćwiczenia 1 oraz 3, które są wyżej omówione. Ponieważ dynamiczną alokację pamięci zastosowałem i omówiłem już w case:3 (zadanie 5 w instrukcji) to tutaj wykorzystam już wcześniej napisane funkcje *DisplayMatrixw*, *DisplayVecw*, *gaussw*, które są dostosowane do dynamicznie zalokowanych tablic. Dodatkowo potrzebuję funkcji *HilbertMatrixw(n, M)*, *ComputeVecw(n, M, b)*, które będą tworzyły macierz współczynników wg algorytmu podanego w instrukcji oraz zdefiniują wektor wyrazów

wolnych. Schemat programu jest identyczny z tym co mam w *case 1*; jedyna zmiana to alokacja pamięci. Tutaj użytkownik musi podać z klawiatury liczbę równań  $n$ .

```
F:\WOJTEK\STUDIA\SEM2\informatyka 2\lab6\Debug\lab6.exe
1 - rozwiązanie metoda Gaussa
2 - obliczanie macierzy trojkatnej oraz wyznacznika
3 - rozwiązanie układu dla wczytanej macierzy (dynamiczna alokacja)
4 - rozwiązanie metoda Gaussa przy wykorzystaniu dynamicznej alokacji pamięci
Podaj numer ćwiczenia: 4
Podaj liczbę równań
9
1.000000 0.500000 0.333333 0.250000 0.200000 0.166667 0.142857 0.125000 0.111111
0.500000 0.333333 0.250000 0.200000 0.166667 0.142857 0.125000 0.111111 0.100000
0.333333 0.250000 0.200000 0.166667 0.142857 0.125000 0.111111 0.100000 0.090909
0.250000 0.200000 0.166667 0.142857 0.125000 0.111111 0.100000 0.090909 0.083333
0.200000 0.166667 0.142857 0.125000 0.111111 0.100000 0.090909 0.083333 0.076923
0.166667 0.142857 0.125000 0.111111 0.100000 0.090909 0.083333 0.076923 0.071429
0.142857 0.125000 0.111111 0.100000 0.090909 0.083333 0.076923 0.071429 0.066667
0.125000 0.111111 0.100000 0.090909 0.083333 0.076923 0.071429 0.066667 0.062500
0.111111 0.100000 0.090909 0.083333 0.076923 0.071429 0.066667 0.062500 0.058824

wektor prawych stron
2.828968
1.428968
1.519877
1.269877
1.096800
0.968229
0.868229
0.787872
0.721695

Wyniki dla n=9
1.000000
1.000000
1.000000
1.000002
0.999993
1.000015
0.999983
1.000010
0.999997
```

Wyniki tej części programu oczywiście są identyczne z wynikami ćwiczenia 1 dla tej samej liczby równań.