Problematyka:

- Program ma wyznaczyć $y = A^{-1}x$ w jak najbardziej optymalny sposób dla:

$$\mathbf{A} = \begin{pmatrix} 1.2 & \frac{0.1}{1} & \frac{0.15}{1^2} & & & & & & \\ 0.2 & 1.2 & \frac{0.1}{2} & \frac{0.15}{2^2} & & & & & \\ & 0.2 & 1.2 & \frac{0.1}{3} & \frac{0.15}{3^2} & & & & \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ & & & & & 0.2 & 1.2 & \frac{0.1}{N-2} & \frac{0.15}{(N-2)^2} \\ & & & & & & 0.2 & 1.2 & \frac{0.1}{N-1} \\ & & & & & & & 0.2 & 1.2 \end{pmatrix}$$

  oraz $x = (1, 2,..., N)^T$. Dla N = 124. Oraz obliczyć wyznacznik macierzy A. I złożoność obliczeniowa powinna wynosić O(n)

- Następnie trzeba potraktować N jako zmienną i zmierzyć czas działania programu w funkcji od N i przedstawić to na wykresie. Dla dobrze napisanego programu można spodziewać się liniowej zależności.

Wyjaśnienie programu:

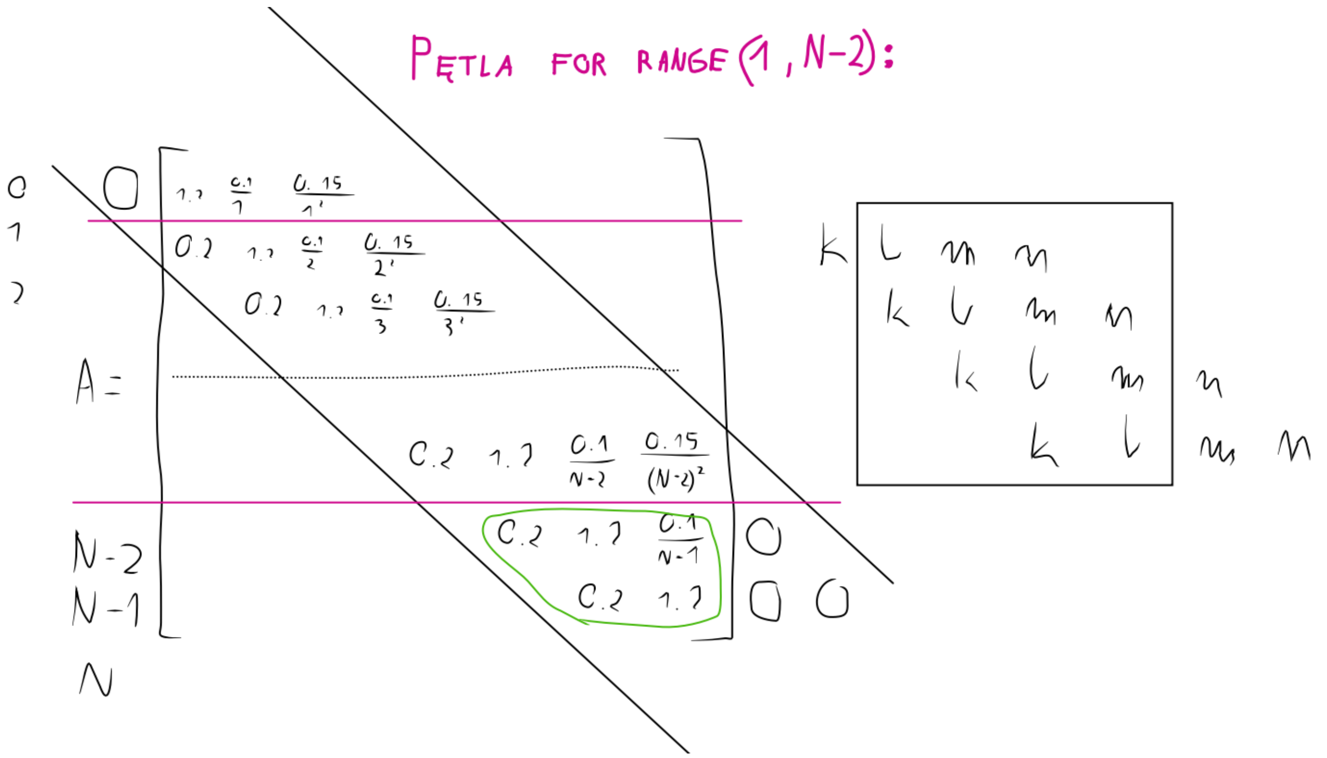
- W programie wykorzystane są funkcję z dodatkowej biblioteki (matplotlib) aby w prosty sposób stworzyć wykres zależności czasu od zmiennej N.

- Funkcję stworzone przeze mnie dla potrzeby rozwiązania problemu:
  - fun_NUM3(n): funkcja przyjmuje liczbę n która określa rozmiar naszej macierzy A oraz wektora x. Zwraca wyliczony wektor y zgodnie z poleceniem zadania oraz wyliczony wyznacznik macierzy A.
  - createPlot(n_values, fun, no_samples): funkcja która tworzy wykres czasu obliczenia funkcji (drugi argument) od wartości N (pierwszy argument) i aby nie było dużych szumów to dla każdej wartości n jest obliczany, podaną ilość razy, czas a później jest liczona średnia ze wszystkich prób.

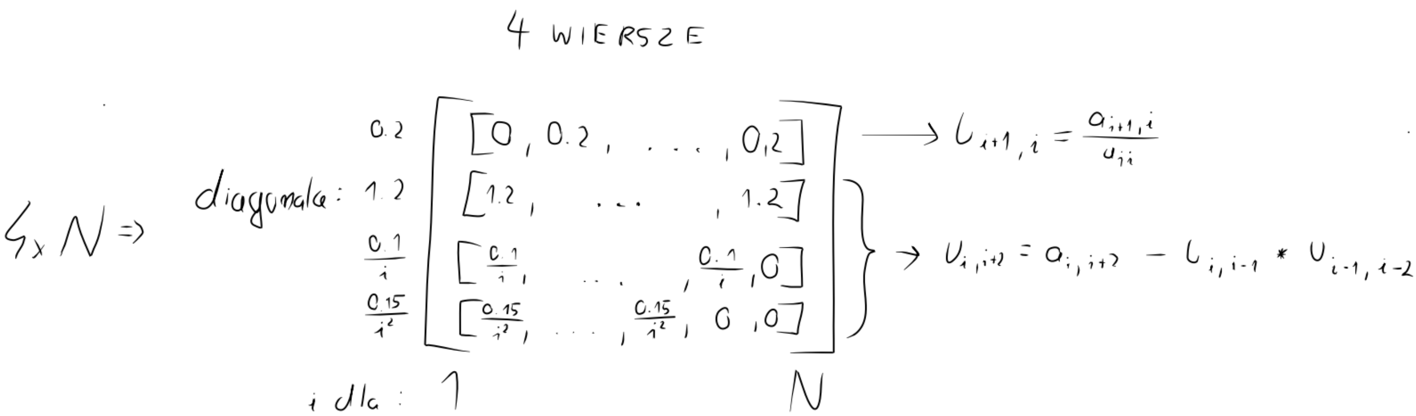
Mateusz Kałwa

Wyjaśnienie problematyki:

zdj 1:



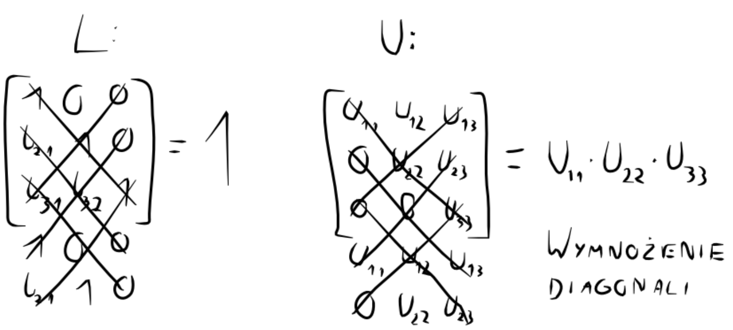
zdj 2:



zdj 3:

WYZNACZNIK:

$$W(A) = W(L) \cdot W(U)$$



zdj 4:

$$y = A^{-1}x \quad | \cdot A_L$$

$$Ay = x$$

$\Downarrow$ METODA LU

$$Lz = x$$

$$Uy = z$$

Mateusz Kałwa

zdj 5:

$$Lz = x:$$

$$\begin{array}{cc} & \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \end{bmatrix} \\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ l_1 & 1 & 0 & 0 & 0 & 0 \\ 0 & l_2 & 1 & 0 & 0 & 0 \\ 0 & 0 & l_3 & 1 & 0 & 0 \\ 0 & 0 & 0 & l_4 & 1 & 0 \\ 0 & 0 & 0 & 0 & l_5 & 1 \end{bmatrix} & \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{bmatrix} \end{array}$$

$$\begin{aligned} 1 &= z_1 & &\Rightarrow z_1 = 1 \\ 2 &= l_1 \cdot z_1 + z_2 & &\Rightarrow z_2 = 2 - l_1 \cdot z_1 \\ 3 &= l_2 \cdot z_2 + z_3 & &\Rightarrow z_3 = 3 - l_2 \cdot z_2 \\ 4 &= l_3 \cdot z_3 + z_4 & &\Rightarrow z_4 = 4 - l_3 \cdot z_3 \\ 5 &= l_4 \cdot z_4 + z_5 & &\Rightarrow z_5 = 5 - l_4 \cdot z_6 \\ 6 &= l_5 \cdot z_5 + z_6 & &\Rightarrow z_6 = 6 - l_5 \cdot z_5 \end{aligned}$$

$$\Downarrow$$

$$x \Rightarrow z$$

$$x[0] = z[0] = 1$$ NIE ZMIENIAMY

NASZA MACIERZ L JEST W 0 INDEKSIE MACIERZY A

DLA $i \in \langle 1, N)$ $$x[i] = x[i] - A[0][i] \cdot x[i-1]$$

zdj 6:

$$Uy = z$$

$$\begin{array}{cc} & \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \end{bmatrix} \\ \begin{bmatrix} U_{10} & U_{20} & U_{30} & 0 & 0 & 0 \\ 0 & U_{11} & U_{21} & U_{31} & 0 & 0 \\ 0 & 0 & U_{12} & U_{22} & U_{32} & 0 \\ 0 & 0 & 0 & U_{13} & U_{23} & U_{33} \\ 0 & 0 & 0 & 0 & U_{14} & U_{24} \\ 0 & 0 & 0 & 0 & 0 & U_{15} \end{bmatrix} & \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \end{bmatrix} \end{array}$$

$$\begin{aligned} z_1 &= U_{10} \cdot y_1 + U_{20} \cdot y_2 + U_{30} \cdot y_3 & & \dots \\ z_2 &= U_{11} \cdot y_2 + U_{21} \cdot y_3 + U_{31} \cdot y_4 & & \\ z_3 &= U_{12} \cdot y_3 + U_{22} \cdot y_4 + U_{32} \cdot y_5 & &\Rightarrow y_3 = (z_3 - (U_{22} \cdot y_4) - (U_{32} \cdot y_5)) / U_{12} \\ z_4 &= U_{13} \cdot y_4 + U_{23} \cdot y_5 + U_{33} \cdot y_6 & &\Rightarrow y_4 = (z_4 - (U_{23} \cdot y_5) - (U_{33} \cdot y_6)) / U_{13} \\ z_5 &= U_{14} \cdot y_5 + U_{24} \cdot y_6 & &\Rightarrow y_5 = (z_5 - (U_{24} \cdot y_6)) / U_{14} \\ z_6 &= U_{15} \cdot y_6 & &\Rightarrow y_6 = z_6 / U_{15} \end{aligned}$$

$$x \Rightarrow z \Rightarrow y$$

$$x[n-1] = x[n-1] / A[1][n-1]$$

$$x[n-2] = (x[n-2] - (A[2][n-2] \cdot x[n-1])) / A[1][n-2]$$

---

dla $i \in \langle n-3, 0 \rangle$ $$x[i] = (x[i] - (A[2][i] \cdot x[i+1]) - (A[3][i] \cdot x[i+2])) / A[1][i]$$

Zacznijmy od wyjaśnienia, dlaczego nasza macierz nie jest zadeklarowana w tablicy NxN tylko w tablicy 4xN oraz dlaczego nie tworzymy osobnych macierzy L i U tylko wszystko trzymamy w jednej zadeklarowanej tablicy.

Do przechowania naszych wartości potrzebujemy zmiennej zmiennoprzecikowej, użyliśmy do tego zmiennej typu float która ma 64 bity. Załóżmy że chcemy obliczyć dany problem dla N = 100 000 co nam daje macierz A: 100000x100000. Czyli potrzebowalibyśmy miejsca w pamięci RAM na 10 000 000 000 zmiennych typu float co daje nam:

640 000 000 000 b = 80 000 000 000 B = 80 000 MB = 80 GB (około)

Tyle potrzebujemy pamięci RAM dla samej jednej macierzy A a jeszcze dodatkowo potrzebne nam są macierze L oraz U, czyli w sumie potrzebowalibyśmy 240 GB pamięci RAM (nie wliczając wektora x).

Teraz popatrzmy na to z innej strony. Widzimy że niezerowe wartości naszej macierzy A “układają” się w 4 wierszach (zdj1 oraz zdj2). I nie potrzebujemy alokować pamięci dla wartości zerowych. Co za tym idzie, możemy stworzyć tablice o rozmiarze 4xN. Załóżmy tak jak w poprzednim przykładzie że N = 100 000 co daje nam tablice 4x100 000. Tym razem potrzebujemy tylko 400 000 zmiennych typu float czyli

25 600 000 b = 3 200 000 B = 3.2 MB (około)

Jak widać różnica w zapotrzebowaniu pamięci jest ogromna. Dodatkowo dzięki odpowiedniemu użyciu metody LU możemy zapisywać po kolei wszystkie wyliczone wartości macierzy L oraz U do naszej już zalokowanej tablicy która na początku posiadała wartości głównej macierzy A.

Wyliczanie macierzy L oraz U dla naszego problemu, na pierwszy rzut oka może wydawać się trudnę, lecz dzięki temu że nasza główna macierz A posiada w określonych miejscach wartości zerowe, nasze obliczenia drastycznie się upraszczają (zdj2). Stosując odpowiednio uproszczone wzory dla odpowiednich indeksów naszej tablicy 4xN, otrzymamy obie macierze L i U zapisane w naszej głównej tablicy 4xN. Trzeba zauważyć że tracimy oryginalne wartości macierzy A, lecz do dalszych obliczeń nie będą nam one już potrzebne.

Obliczenie wektora y po obliczeniu macierzy L oraz U ze wzoru $y = A^{-1}x$ sprowadza się do dwóch układów równań z macierzami trójkątnymi (zdj4). Posiadając wzory na obliczenie wektora y poprzez macierze L oraz U, musimy wyprowadzić wzory dla naszej problematyki na obliczanie naszych układów równań (zdj5 i zdj6). Najważniejsze jest to że uwzględniamy możliwość alokacji tylko jednej tablicy 1xN dla naszego wektora x który po obliczeniach będzie przechowywał obliczone wartości wektora y.

Aby wyliczyć wyznacznik naszej macierzy A w postaci A = LU pomoże nam twierdzenie Cauchy’ego dzięki któremu możemy bardzo uprościć nasze obliczenia (zdj3). Wystarczy że pomnożymy przez siebie wszystkie wartości na naszej diagonali która jest umieszczona w 1 indeksie naszej tablicy 4xN (zdj1 i zdj2).

Posiadając wyprowadzone wszystkie wzory tworzymy funkcję która będzie odliczała nam wszystko. Teraz zostało nam obliczenie czasu potrzebnego funkcji dla danej wartości N i stworzenie wykresu zależności czasu od wartości N. Obliczając tylko raz czas potrzebny na wykonanie funkcji dla wartości N od 124 do 1000 można zauważyć że wykres jest bardzo poszarpany. Aby temu zaradzić wystarczy każdy czas potrzebny dla danej wartości N wyliczać jakąś określoną liczbę razy a później brać średnią wartość czasu dla danego N.

Mateusz Kałwa

Wnioski:

- Nasz program dzięki odpowiednim optymalizacją potrzebuję do funkcjonowania znacznie mniejszej ilości pamięci RAM oraz jego złożoność obliczeniowa jest liniowa O(n).

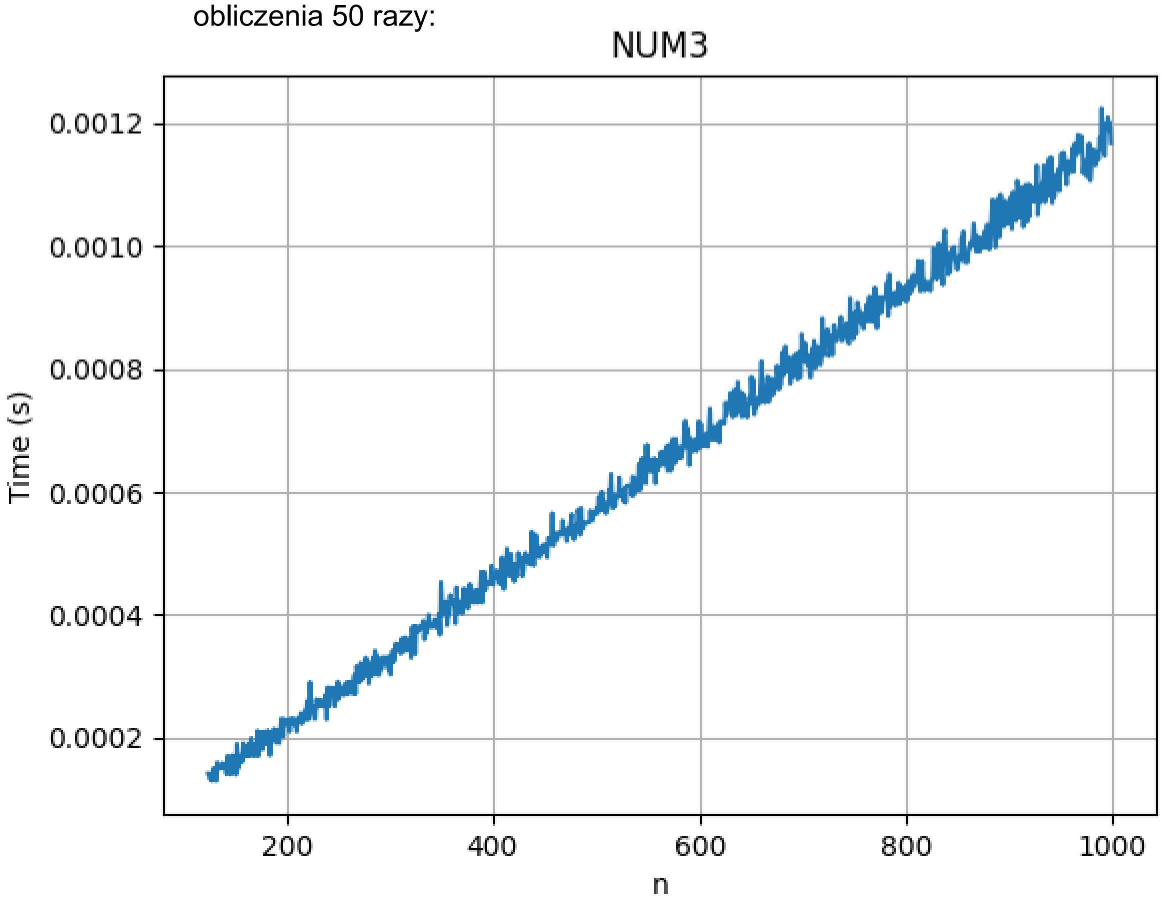
Uruchamianie programu:

- make y : uruchamia program który wypisuje obliczony wektor y
- make time : uruchamia program który tworzy wykres czasu funkcji od wartości n
- make det : uruchamia program który wypisuję wyznacznik macierzy A

W razie problemów można zmienić w pliku make wywołanie pythona z “python” na “python3”

Wyniki programu:

- Wykres zależności czasu od wartości N dla n od 124 do 1000 i próbkowaniu każdego obliczenia 50 razy:



- Obliczony wyznacznik macierzy A dla N = 124:

Calculated determinant of the matrix A:

6141973498.857843

Mateusz Kałwa

- Obliczony wektor y dla N = 124

[0.448700827728733, 1.4132732869357947, 2.1348778535462736, 2.8690132654396248, 3.5914885705595205, 4.311604959915503, 5.029827173723323, 5.747011462584994, 6.463503693914558, 7.179525964548697, 7.8952125968955915, 8.610651859797315, 9.325903619364162, 10.041009954626537, 10.756001271894783, 11.470900080737994, 12.185723394049369, 12.900484305313817, 13.615193053266005, 14.329857758065131, 15.044484941235352, 15.759079899902147, 16.473646980766127, 17.18818978377055, 17.902711315621058, 18.617214106977666, 19.331700302956037, 20.046171733762908, 20.76062997036838, 21.47507636878333, 22.189512105570603, 22.903938206548368, 23.6183555701598, 24.332764986629712, 25.047167153767735, 25.761562690082965, 26.475952145728595, 27.190336011683936, 27.904714727496145, 28.61908868783829, 29.33345824808956, 30.047823729103516, 30.762185421298863, 31.476543588182352, 32.19089846939369, 32.90525028334641, 33.61959922952588, 34.33394549049516, 35.048289233651346, 35.762630612767495, 36.4769697693505, 37.19130683383959, 37.90564192666726, 38.6199751592003, 39.33430663457682, 40.04863644845216, 40.762964689665075, 41.47729144083417, 42.19161677889273, 42.905940775569235, 43.62026349782013, 44.33458500822004, 45.04890536531427, 45.763224623938, 46.47754283550541, 47.19186004827236, 47.90617630757509, 48.62049165604775, 49.334806133820685, 50.04911977870149, 50.76343262634067, 51.47774471038327, 52.192056062607854, 52.9063667130542, 53.62067669014054, 54.33498602077156, 55.04929473043772, 55.76360284330703, 56.47791038230969, 57.192217369216245, 57.906523824710035, 58.62082976845425, 59.335135219153926, 60.049440194613666, 60.763744711791205, 61.47804878684707, 62.192352435190934, 62.90665567152471, 63.62095850988271, 64.3352609636691, 65.04956304569288, 65.76386476820053, 66.47816614290662, 67.19246718102224, 67.90676789328191, 68.62106828996849, 69.33536838093679, 70.0496681756356, 70.76396768312829, 71.47826691211242, 72.19256587093781, 72.90686456762393, 73.62116300987596, 74.33546120510012, 75.04975916041795, 75.76405688267984, 76.47835437847795, 77.19265165415811, 77.90694871583132, 78.62124556938441, 79.33554222049035, 80.04983867461782, 80.76413493704023, 81.47843101284448, 82.19272690693916, 82.90702262406211, 83.62131816878798, 84.33561354553508, 85.04990875857204, 85.76420381203626, 86.47849870460276, 87.19279247126808, 87.90778524137869, 88.68203579310355]