Sprawozdanie z zadania numerycznego 3

1. Instrukcja uruchomienia

- Aby uruchomić program należy użyć *python NUM3.py*

2. Cel ćwiczenia

Efektywne wyznaczenie $y = A^{-1}x$, gdzie A jest macierzą rzadką mającą niezerowe elementy na diagonali, pod diagonali i dwóch nad diagonalach.

3. Opis ćwiczenia

1) Przekształcenie polecenia i wyznaczenie wzorów
2) Rozkład LU macierzy A
3) Podstawianie w przód
4) Podstawianie wstecz
5) Obliczenie wyznacznika

4. Wstęp teoretyczny

Zacznijmy od omówienia problemu przedstawionego w zadaniu. Z polecenia wynika że mając macierz A musimy obliczyć do niej macierz odwrotną. Na ogół jest to operacja o dużej złożoności obliczeniowej i podczas rozwiązywania problemów numerycznych powinno się takich przekształceń unikać. Dlatego nasz wzór z polecenia wystarczy przekształcić mnożąc obustronnie po lewej stronie równanie przez macierz A.

$$y = A^{-1}x \Leftrightarrow Ay = x$$

Mając równanie w takiej postaci przy zastosowaniu odpowiednich algorytmów można znacznie przyspieszyć czas obliczeń.

Aby wybrać odpowiedni algorytm trzeba przyjrzeć się strukturze macierzy A. Jest to macierz mająca zapełnione tylko cztery diagonale. Z tego faktu można wyciągnąć dwa wnioski:

- Bardzo nieefektywne jest przechowywanie takiej macierzy o n elementach w tablicy n*n, gdy i tak większość elementów to są 0, które zajmują pamięć, a do niczego się nie przydają.
- Sprowadzenie naszej macierzy do macierzy trójkątnej pozwoliło by na zastosowanie forward lub backward substitution do wyliczenia wyniku. Obie te metody mają ogólną złożoność czasową $O(n^2)$ bo muszą przejść przez n wierszy macierzy, w których liczba kolumn po których trzeba przejść rośnie do n.

Do przechowywania diagonali możemy użyć macierzy wstęgowej, którą w tym zadaniu stworzyłem poprzez listę zawierającą w sobie 4 listy 100 elementowe (są to nasze 4 diagonale). Macierz ma postać:

$$\begin{pmatrix} 0 & x & x & x \\ x & x & x & x \\ \vdots & \vdots & \vdots & \vdots \\ x & x & x & x \\ x & x & x & 0 \\ x & x & 0 & 0 \end{pmatrix}$$

Rozkład który wybrałem do rozłożenia naszej macierzy A na dwie macierze trójkątne to LU. Ma on w ogólnym przypadku złożoność czasową $O(n^3)$, lecz uwzględniając to jaką macierz będziemy rozkładać możemy to zrobić w czasie liniowym O(n). Rozkład polega na rozbiciu naszej macierzy na dwie macierze trójkątne (górną i dolną).

Algorytm którego użyłem jest algorytm Dollittle'a którego wzory ogólne wyglądają następująco:

$$U_{ij} = A_{ij} - \sum_{k<i} L_{ik} U_{kj}$$

$$L_{ij} = \frac{A_{ij} - \sum_{k<j} L_{ik} U_{kj}}{U_{jj}}$$

W naszym przypadku zakładając że elementem diagonalnym jest element U_{ii} musimy posiadać wzory tylko na elementy $U_{ii}, U_{i,i+1}, U_{i,i+2}, L_{i+1,i}$

$$U_{ii} = A_{ii} - \sum_{k<i} L_{ik} U_{ki} = A_{ii} - L_{i,i-1} U_{i-1,i} - L_{i,i-2} U_{i-2,i} - \ldots = A_{ii} - L_{i,i-1} U_{i-1,i}$$

$$U_{i,i+1} = A_{i,i+1} - \sum_{k<i} L_{ik} U_{k,i+1} = A_{i,i+1} - L_{i,i-1} U_{i-1,i+1} - L_{i,i-2} U_{i-2,i+1} - \ldots = A_{i,i+1} - L_{i,i-1} U_{i-1,i+1}$$

$$U_{i,i+2} = A_{i,i+2} - \sum_{k<i} L_{ik} U_{k,i+2} = A_{i,i+2} - L_{i,i-1} U_{i-1,i+2} - \ldots = A_{i,i+1}$$

$$L_{i+1,i} = \frac{A_{i+1,i} - \sum_{k<j} L_{i+1,k} U_{ki}}{U_{ii}} = \frac{A_{i+1,i} - L_{i+1,i-1} U_{i-1,i} - \ldots}{U_{ii}} = \frac{A_{i+1,i}}{U_{ii}}$$

Fragmenty zaznaczone na czerwono przy naszej macierzy wypełnionej w większości zerami są zerami i znacznie upraszczają nam wzory.

Następnie z faktu że A = LU oraz Ax = y otrzymuję równanie:

$$LUy = x$$

Które rozbijam na dwa równania:

$$Lz = x$$ (rozwiązywane przez forward substitution)

$$Uy = z$$ (rozwiązywane przez backward substitution)

Rozwiązanie tych równań jest również czasu O(n) z powodu tego jak zbudowane są nasze macierze trójkątne. Aby łatwo można było wyliczyć te równania wyprowadziłem wzory.
Dla forward substitution:

$$z_0 = x_0$$

$$z_n = x_n - L_n z_{n-1}$$

Dla backward substitution wprowadzę oznaczenia U_0, U_1, U_2 które odpowiadają diagonali, i dwóm nad diagonalom macierzy U:

$$y_{99} = \frac{z_{99}}{U_{0_{99}}}$$

$$y_{98} = \frac{z_{98} - U_{1_{98}} y_{99}}{U_{0_{98}}}$$

Dla n < 98: $$y_n = \frac{z_n - U_{1_n} y_{n+1} - U_{2_n} y_{n+2}}{U_{0_n}}$$

5. Wyniki

Rozwiązaniem równania jest wektor:

y = [0.03287133486041395, 1.3396227980963753, 2.066480295894664, 2.825543605175336, 3.557571715528883, 4.284492868897645, 5.00721018451999, 5.727664002754518, 6.446615582748809, 7.164554400995276, 7.881773878242026, 8.598465868371878, 9.314759799907844, 10.030746230199034, 10.74649032115277, 11.462040127963592, 12.177431844626687, 12.892693237901542, 13.60784595684208, 14.322907124390252, 15.03789045794619, 15.75280707355121, 16.467666073000725, 17.182474979167374, 17.897240063340146, 18.611966594532937, 19.32665903159678, 20.041321172855753, 20.75595627381683, 21.47056714061568, 22.185156204831525, 22.899725583859315, 23.61427712998635, 24.328812470561147, 25.043333041083297, 25.757840112626393, 26.472334814693667, 27.186818154368854, 27.901291032443737, 28.615754257064278, 29.33020855532933, 30.04465458319117, 30.75909293394065, 31.473524145507586, 32.18794870676451, 32.902367062989086, 33.61677962061327, 34.331186751365145, 35.04558879589254, 35.75998606694211, 36.47437885215638, 37.18876741654113, 37.90315200464761, 38.617532842507245, 39.331910139350974, 40.04628408914067, 40.7606548719361, 41.47502265511775, 42.189387594482916, 42.90374983523002, 43.6181095128443, 44.33246675389621, 45.04682167676243, 45.76117439227791, 46.47552500432681, 47.189873610378676, 47.904220301975755, 48.61856516517662, 49.33290828096055, 50.047249725596565, 50.761589570980924, 51.47592788494589, 52.19026473154275, 52.904600171301595, 53.61893426146981, 54.33326705623165, 55.04759860691019, 55.761928962153874, 56.47625816810818, 57.19058626857465, 57.90491330515779, 58.61923931740096, 59.33356434291259, 60.04788841748285, 60.76221157519233, 61.47653384851288, 62.1908552684013, 62.9051758643867, 63.61949566465193, 64.33381469610926, 65.04813298447127, 65.76245055431694, 66.47676742915336, 67.19108363147355, 67.9053991828134, 68.61971410401006, 69.33402833257784, 70.04833794418792, 70.7650588638003, 71.53915685603329]^T

Wartość wyznacznika macierzy A wynosi: 78240161.00959387

6. Wnioski

Wyniki programu są prawie identyczne w porównaniu do wyników które otrzymałem obliczając to samo zadanie z użyciem biblioteki Numpy. Różnicę w tych wynikach można zauważyć dopiero na 15 miejscu po przecinku co prawdopodobnie spowodowane jest błędem wynikającym z precyzji obliczeń.

Własnoręczna implementacja działa też znacznie krócej w porównaniu do czasu pracy metod z biblioteki Numpy.