

# Program do rozwiązywania układów równań liniowych metodą Choleskiego



# Opis programu

- Program rozwiązuje jednorazowo jedno równanie
- Równanie jest wprowadzane przez użytkownika za pomocą GUI
- Elementy GUI zostały napisane w języku C# z wykorzystaniem elementów WPF
- Program posiada dwie biblioteki DLL:
  - W języku wysokiego poziomu – język C
  - W języku niskiego poziomu – assemblerze
- Program stworzony został w Visual Studio 15 dla komputerów wyposażonych w 64 bitowy procesor oraz z systemem Windows 10

# Wzory opisujące metodę Choleskiego

Wzory opisujące metodę obliczeń, zakładając układ opisany równaniem **AX=B**:

$$A=LU$$

$$\text{gdzie } L = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ l_{21} & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & \cdots & 1 \end{bmatrix} \quad U = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u_{nn} \end{bmatrix}.$$

Macierze L i U liczymy według poniższych wzorów:

$$l_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}} \quad k=1, 2, \dots, n \quad i=k+1, \dots, n.$$

$$u_{kk} = a_{kk}^{(k)} \quad k=1, 2, \dots, n \quad i=k, k+1, \dots, n.$$

Algorytm rozwiązania układu równań  $AX=B$  jest następujący – rozbijając równanie  $LUX=B$  otrzymujemy dwa równania  $LY=B$  i  $UX=Y$ . Z pierwszego równania wyznaczamy wektor Y a z drugiego wektor X, według poniższych wzorów:

$$\begin{cases} y_1 = b_1 \\ y_i = b_i - \sum_{k=1}^{i-1} l_{ik} y_k \quad i = 2, 3, \dots, n \end{cases}$$

$$\begin{cases} x_n = \frac{y_n}{u_{nn}} \\ x_i = \frac{y_i - \sum_{k=i+1}^n u_{ik} x_k}{u_{ii}} \quad i = n-1, n-2, \dots, 1 \end{cases}$$

# Interfejs użytkownika

Pole do wprowadzenia ilości wiadomych

Tabela do wprowadzenia danych

Pola do wyboru trybu działania programu

Czas wykonania bibliotek

Podaj liczbę wiadomych

11

Zatwierdź

☐ DLL ASM

☐ DLL C

☒ Test porównawczy

Licz

Czas wykonania

Czas wykonania ASM:

0,0009ms

Czas wykonania C:

0,0012ms

x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	y
-6101	1593	1958	1610	3176	1531	7520	8709	3671	4265	-9401	2791
2601	9195	-6731	-1057	0	-7718	-1605	-6326	1137	1052	2860	4349
4361	1534	2051	5461	1569	4255	5376	7479	4646	1373	8972	5455
3058	2985	2084	-1641	-7645	9910	-1221	6928	7497	-7006	-1160	1011
2225	-3775	3413	2818	1623	1032	-1425	-7855	9340	5036	1575	2597
-2394	2122	3868	8063	3386	8431	2843	1012	2507	1776	3629	4681
3495	-1294	6850	-1551	3912	2955	1753	-4687	1000	-4414	4178	5708
-1320	6660	1991	1358	4394	-5868	4696	7213	-8183	-4841	8398	1757
-2953	9796	1899	1903	2694	-9455	1751	-5598	1130	-6572	4460	2936
9389	8751	1498	-2002	3204	2751	1231	2743	-3440	2140	-1507	4092
3671	2084	2904	2755	3105	1135	-4740	5342	7739	1725	2901	2535

Wynik

x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10
-0.15	0.20	0.44	-0.46	-0.55	0.53	0.14	0.02	-0.20	0.43	0.61

Tabela z wynikiem

# Biblioteki DLL

- ▶ Biblioteka wysokiego poziomu w języku C
- ▶ Biblioteka w języku Asembler
- ▶ Realizują obliczanie wektora wynikowego (macierze L i U są tworzone w programie).
- ▶ Są dołączane dynamicznie
- ▶ Nie wspierają wielowątkowości

# Podsumowanie

- ▶ Udało się zrealizować wszystkie założenia projektu
- ▶ Biblioteka w asemblerze działa szybciej niż biblioteka w języku C

Dziękuję za uwagę