**Лабораторна робота №4**

**Розв’язання систем лінійних алгебраїчних рівнянь.**

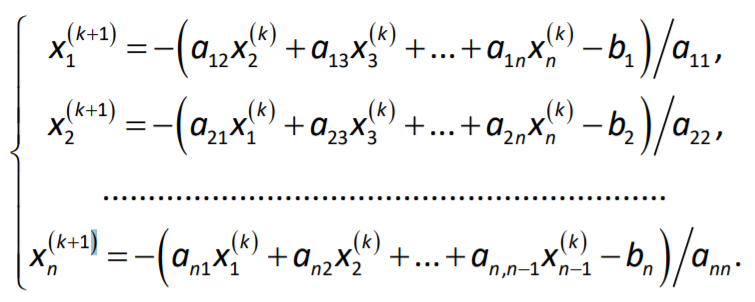
**Ітераційні методи Якобі та Гауса-Зейделя**

10 варіант

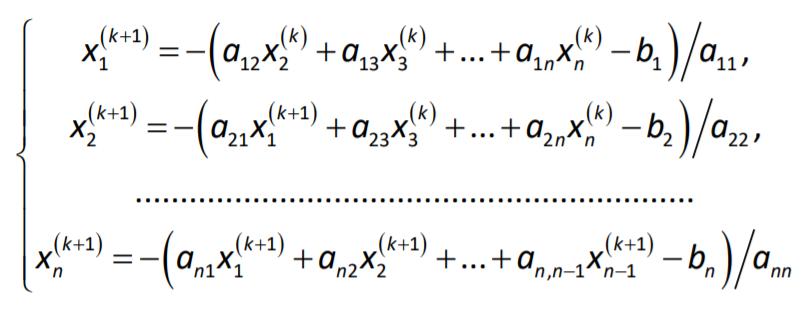
**ЗАВДАННЯ**

З’ясувати факт збіжності чи розбіжності ітераційних процесів Якобі та Зейделя. У випадку збіжності знайти розв’язок СЛАР з точністю 0.0001 та перевірити його, підставляючи в СЛАР отримані розв’язки і обраховуючи нев’язки. Визначити порядок збіжності ітераційного процесу.

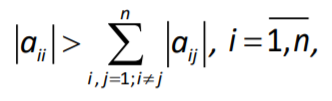
Розрахункові формули:

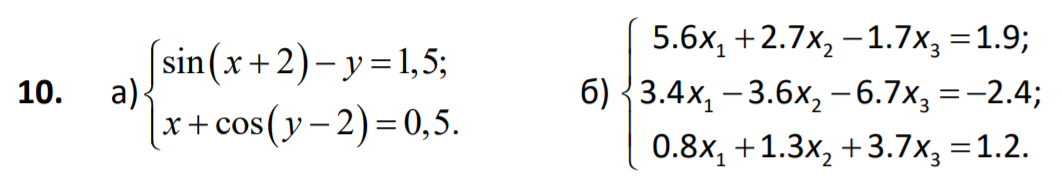
Перетворення СЛАР методом Якобі:

Перетворення СЛАР методом Зейделя:



Достатня умова збіжності (умова діагональної переваги):



Варіант: 

КОД

using System;

using System.Collections.Generic;

using System.Linq;

using System.Collections.Generic;

using System.Diagnostics;

using System.Text;

using System.Threading.Tasks;

namespace program4

{

class Program

{

static double X(double y, double x)

{

return (x + Math.Cos(y - 2.0)) / 1.2;

}

static double Y(double x)

{

return 1.5 - Math.Sin(x + 2);

}

static double X1(double x2, double x3)

{

double x1 = (-2.7 \* x2 + 1.7 \* x3 + 1.9) / 5.6;

return x1;

}

static double X2(double x1, double x3)

{

double x2 = (-3.4 \* x1 + 6.7 \* x3 - 2.4) / -3.6;

return x2;

}

static double X3(double x1, double x2)

{

double x3 = (-0.8 \* x1 - 1.3 \* x2 + 1.2) / 3.7;

return x3;

}

static void Answer(double x, double y)

{

double xz = x;

double yz = y;

double xk = x;

double yk = y;

int iter = 1;

double d = Math.Max(Math.Abs(x - X(y, x)), Math.Abs(y - Y(x)));

Console.WriteLine("МПI");

if (d <= 0.0001) { }

else

{

Console.WriteLine("N\t\ta\t\tb \t\td");

for (; d > 0.0001; iter++)

{

if (iter <= 100)

{

xk = X(y, x);

yk = Y(x);

d = Math.Max(Math.Abs(x - xk), Math.Abs(y - yk));

Console.WriteLine("{0}\t\t{1:f5}\t\t{2:f5} \t\t{3:f5}", iter, xk, yk, d);

x = xk;

y = yk;

}

else { d = 0; Console.WriteLine("Iтерацiйний процес розбiгається"); }

}

}

if (iter <= 100)

{

Console.WriteLine("X = " + xk + "Y = " + yk);

Console.WriteLine("Вектор невязки:" + (Math.Cos(xk) + yk - 1.2) + ";" + (2 \* xk - Math.Sin(yk - 0.5) - 2));

iter = 0;

Console.WriteLine("Метод Зейделя");

Console.WriteLine("№ \t\t x1 \t\t\tx2 \t\t\tПохибка");

double prex;

double prey;

do

{

iter++;

prex = xz;

prey = yz;

xz = X(yz, xz);

yz = Y(xz);

Console.WriteLine($"{iter}\t \t {xz:f5} \t \t{yz:f5} \t\t{Math.Max(Math.Abs(prex - xz), Math.Abs(prey - yz))}");

} while (Math.Abs(prex - xz) > 0.0001 || Math.Abs(prey - yz) > 0.0001);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.WriteLine($"X: {xz}");

Console.WriteLine($"Y: {yz}");

Console.WriteLine($"Вектор невязки по першому рiвнянню: {Math.Sin(xz + 2) - yz - 1.5}");

Console.WriteLine($"Вектор невязки по другому рiвнянню: {xz + Math.Cos(yz - 2) - 0.5}");

}

}

static void Main(string[] args)

{

try

{

Console.WriteLine("Input X"); double x = double.Parse(Console.ReadLine());

Console.WriteLine("Input Y"); double y = double.Parse(Console.ReadLine());

Answer(x, y);

int i = 0;

double x1 = 1, x2 = 0.5, x3 = 0;

double nextx1, nextx2, nextx3;

Console.WriteLine("Метод Якобi");

Console.WriteLine("№ \t\t x1 \t\t\tx2 \t\t\tx3 \t\t\tПохибка");

do

{

i++;

nextx1 = x1;

nextx2 = x2;

nextx3 = x3;

x1 = X1(nextx2, nextx3);

x2 = X2(nextx1, nextx3);

x3 = X3(nextx1, nextx2);

Console.WriteLine($"{i}\t \t {x1:f5} \t \t{x2:f5} \t\t{x3:f5} \t\t{Math.Max(Math.Abs(nextx1 - x1), Math.Max(Math.Abs(nextx2 - x2), Math.Abs(nextx3 - x3)))}");

} while (Math.Abs(nextx1 - x1) > 0.0001 || Math.Abs(nextx2 - x2) > 0.0001 || Math.Abs(nextx3 - x3) > 0.0001);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.WriteLine($"x1: {x1}");

Console.WriteLine($"x2: {x2}");

Console.WriteLine($"x3: {x3}");

Console.WriteLine($"Вектор невязки по першому рiвнянню: {-1.9 + 5.6 \* x1 + 2.7 \* x2 - 1.7 \* x3}");

Console.WriteLine($"Вектор невязки по другому рiвнянню: {2.4 + 3.4 \* x1 - 3.6 \* x2 - 6.7 \* x3}");

Console.WriteLine($"Вектор невязки по третьому рiвнянню: {-1.2 + 0.8 \* x1 + 1.3 \* x2 + 3.7 \* x3}");

x1 = 1; x2 = 0.5; x3 = 0;

Console.WriteLine();

i = 0;

Console.WriteLine("Метод Зейделя (СЛАР):");

Console.WriteLine("№ \t\t x1 \t\t\tx2 \t\t\tx3 \t\t\tПохибка");

do

{

i++;

nextx1 = x1;

nextx2 = x2;

nextx3 = x3;

x1 = X1(x2, x3);

x2 = X2(x1, x3);

x3 = X3(x1, x2);

Console.WriteLine($"{i}\t \t {x1:f5} \t \t{x2:f5} \t\t{x3:f5} \t\t{Math.Max(Math.Abs(nextx1 - x1), Math.Max(Math.Abs(nextx2 - x2), Math.Abs(nextx3 - x3)))}");

} while (Math.Abs(nextx1 - x1) > 0.0001 && Math.Abs(nextx2 - x2) > 0.0001 && Math.Abs(nextx3 - x3) > 0.0001);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.WriteLine($"x1: {x1}");

Console.WriteLine($"x2: {x2}");

Console.WriteLine($"x3: {x3}");

Console.WriteLine($"Вектор невязки по першому рiвнянню: {-1.9 + 5.6 \* x1 + 2.7 \* x2 - 1.7 \* x3}");

Console.WriteLine($"Вектор невязки по другому рiвнянню: {2.4 + 3.4 \* x1 - 3.6 \* x2 - 6.7 \* x3}");

Console.WriteLine($"Вектор невязки по третьому рiвнянню: {-1.2 + 0.8 \* x1 + 1.3 \* x2 + 3.7 \* x3}");

Console.ReadKey();

}

catch (Exception e)

{

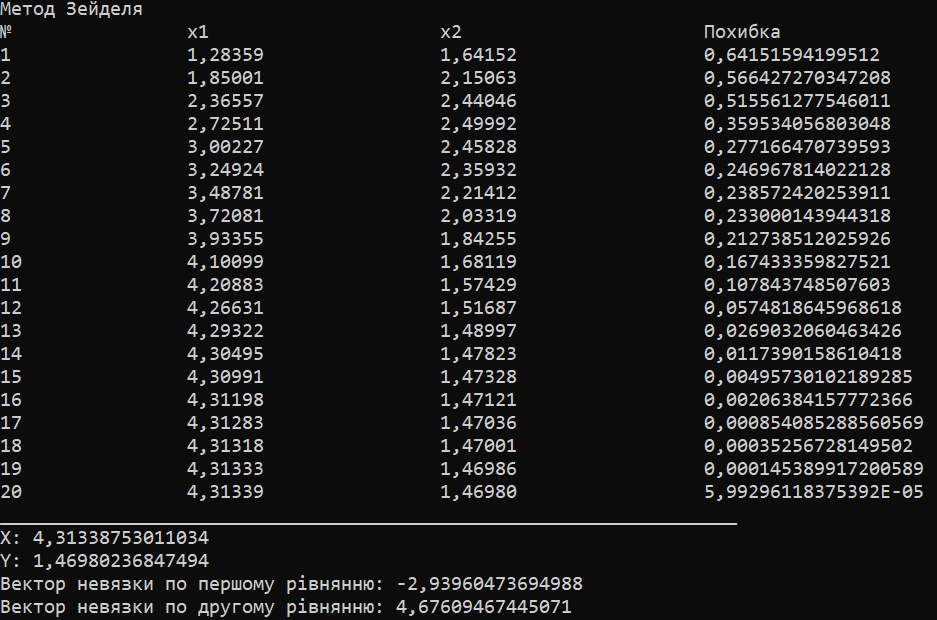
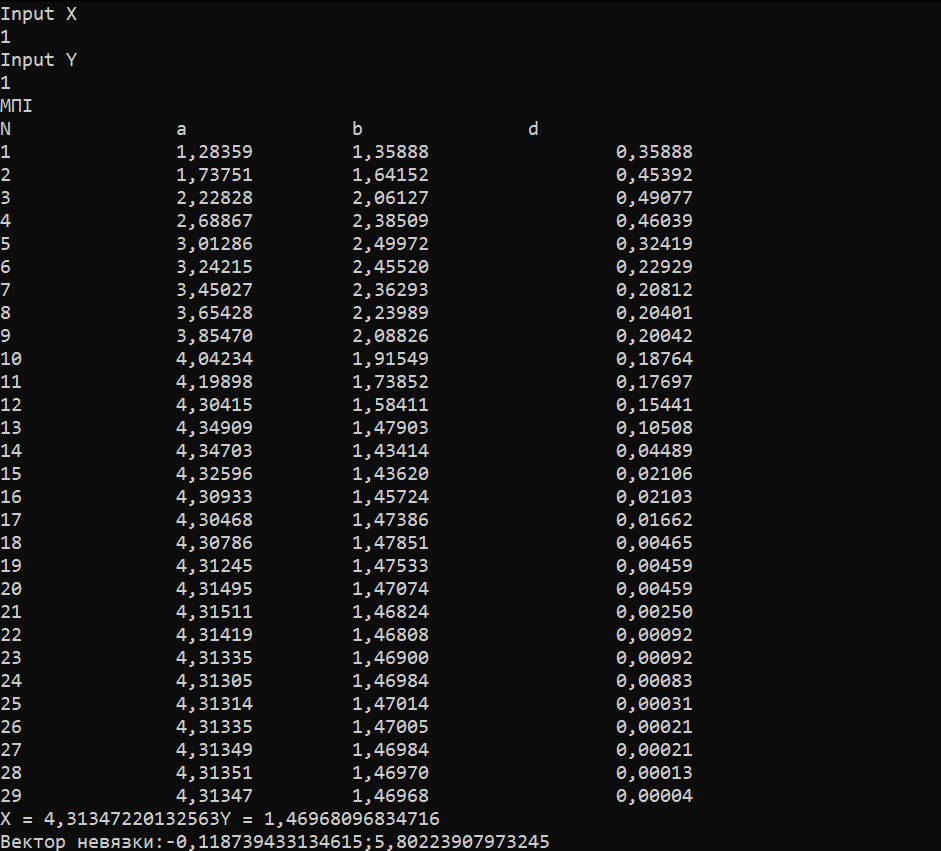
Console.WriteLine(e.Message);

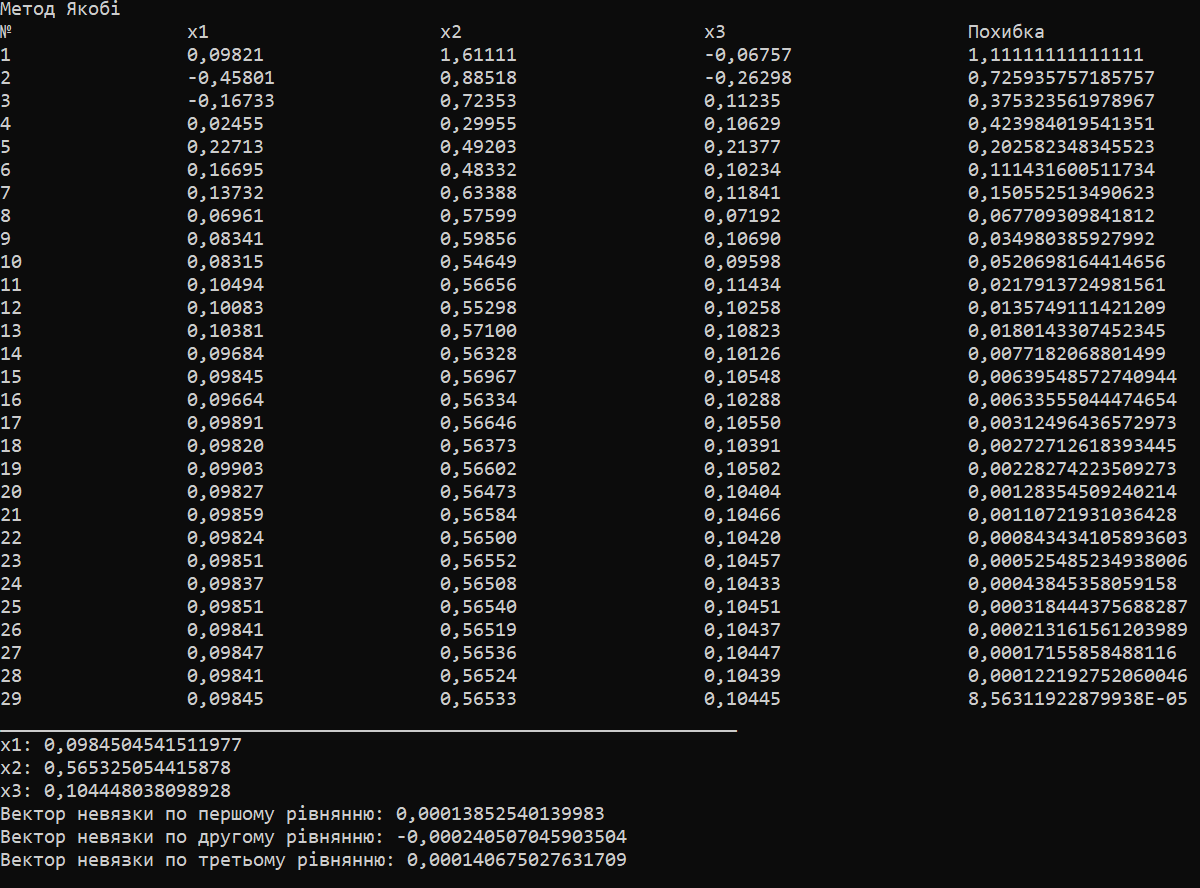
}

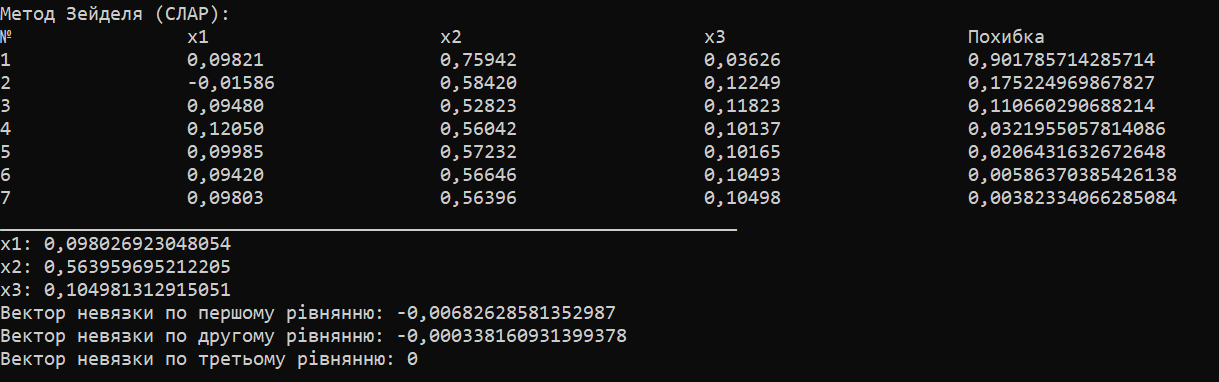
}

}

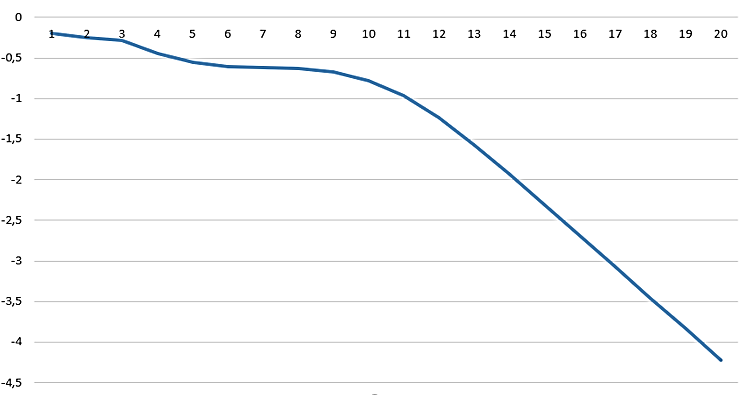
}



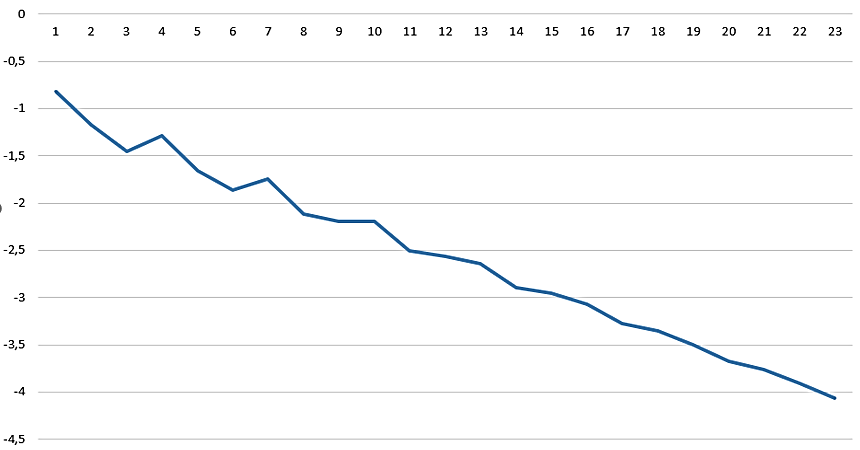




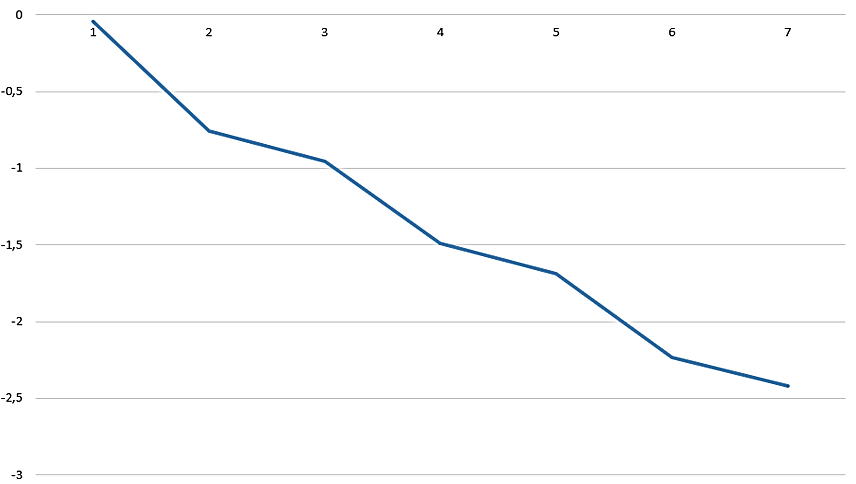
1. ***Зейделя***



1. ***Метод Якобі***



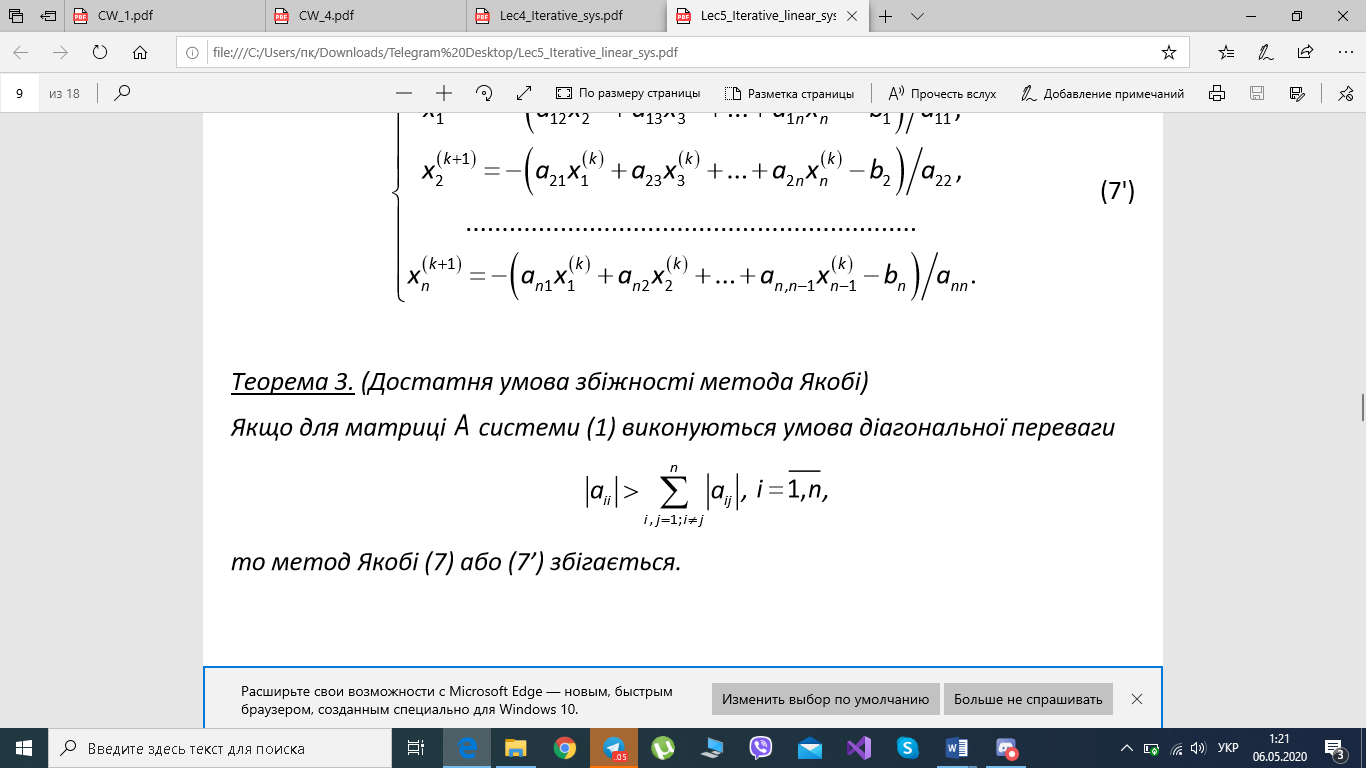
1. ***Метод Зейделя (система лінійних алгебраїчних рівнянь)***



КОНТРОЛЬНІ ПИТАННЯ

1. ***Сформулюйте достатні умови збіжності методів простих ітерацій та Якобі.***

Кубічна норма матриці Якобіана повинна бути менша за одиницю.



1. ***Які ви знаєте способи вибору початкового наближення для розв’язання системи рівнянь ітераційним методом?***

Якщо видно відразу в яких межах буде корінь, то і початкове наближення обирається в тих самих межах

1. ***Які з норм вектора можливо і доцільно використовувати в умовах закінчення ітерацій?***

Октаедрична , евклідова,кубічна норми

1. ***Як попередньо оцінити кількість необхідних для розв’язання СЛАР ітерацій? Яке початкове наближення розв’язку доцільно використати при цьому?***

Можна знайти за Апостеорною і Апріорною похибкою. Нерівність працює в обидва боки.

1. ***Як зміниться умова закінчення ітерацій, якщо у якості норми вектора взяти першу та другу норми?***

Збільшиться кількість ітерацій .особливо в евклідовій нормі , бо, коли корінь буде добуватися мало чим різних значень , то і похибка буде більшою.

ВИСНОВОК

***Протягом лабораторної роботи було вивчено алгоритми для розв’язання систем ітераційними методами: метод простих ітерацій, Якобі, Зейделя.***