Національний технічний університет України «КПІ ім. Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра інформаційних систем та технологій

Лабораторна робота № 3

з дисципліни «Спеціальні розділи математики-2.  
Чисельні методи»

на тему

«**Розв’язання систем лінійних алгебраїчних рівнянь (СЛАР) ітераційними методами. Метод простої ітерації. Метод Зейделя**»

Виконав:

студент гр. ІС-34

Колосов Ігор

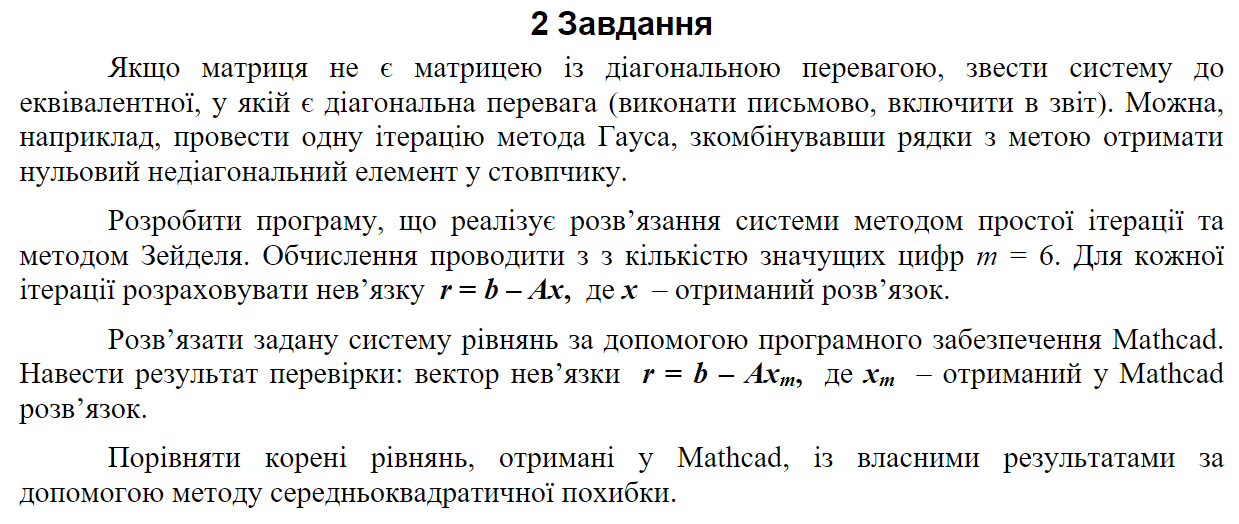
Викладач:

доц. Рибачук Л.В.

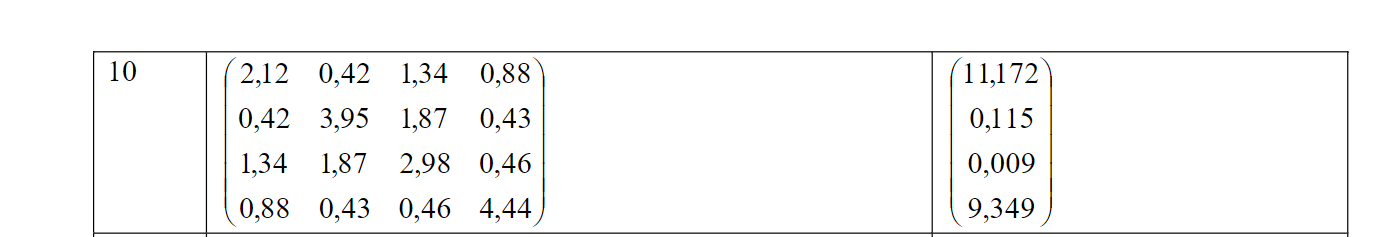
Київ – 2024

**Зміст**

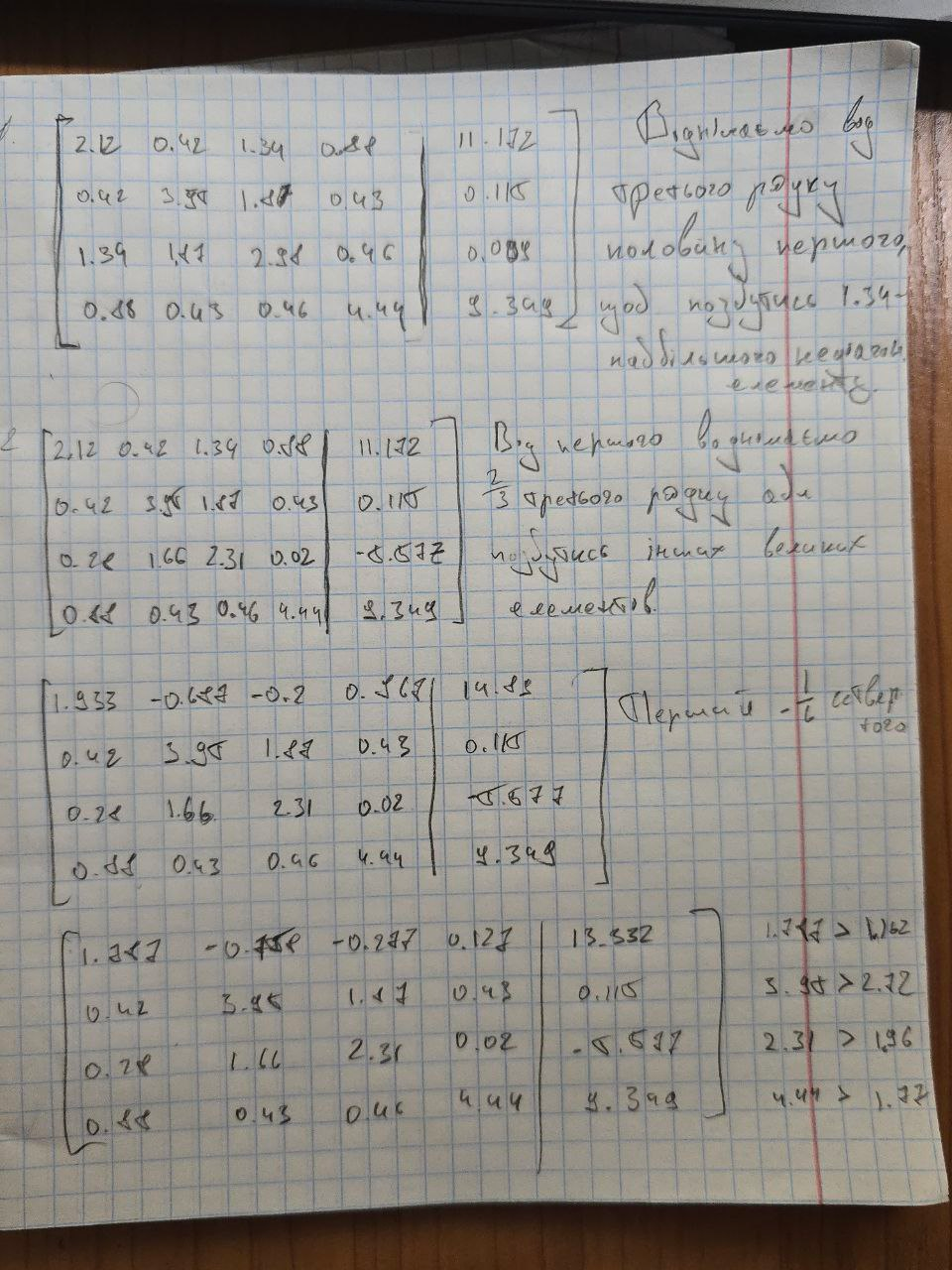
1. **Постановка Задачі**

****

1. **Вихідна система рівнянь**

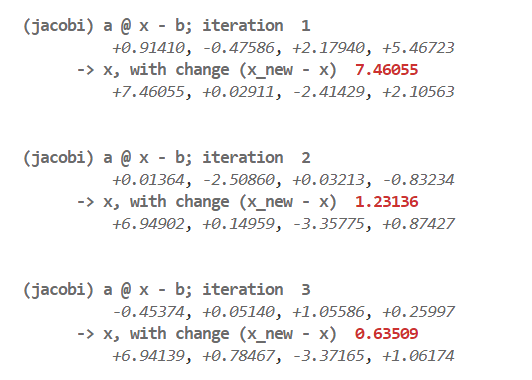
****

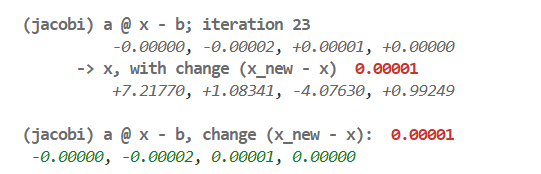
1. **Письмовий етап приведення матриці до діагональної переваги**

****

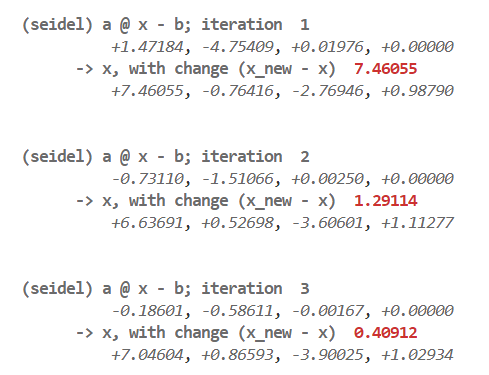
1. **Результати трьох та останньої ітерації методу (Із вектором нев’язки)**

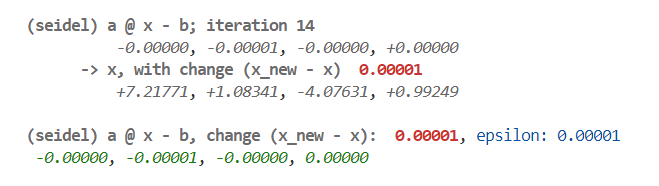
* **Метод Якобі**

****

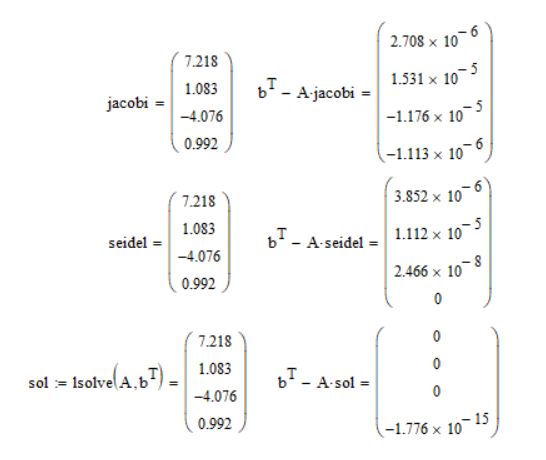
****

* **Метод Зейделя**

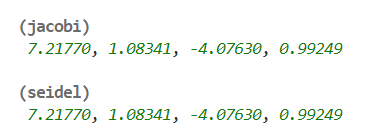
****

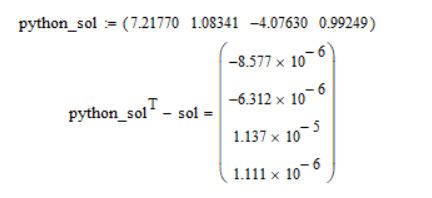
****

1. **Копія розв’язку задачі у Mathcad; Вектор нев’язки**

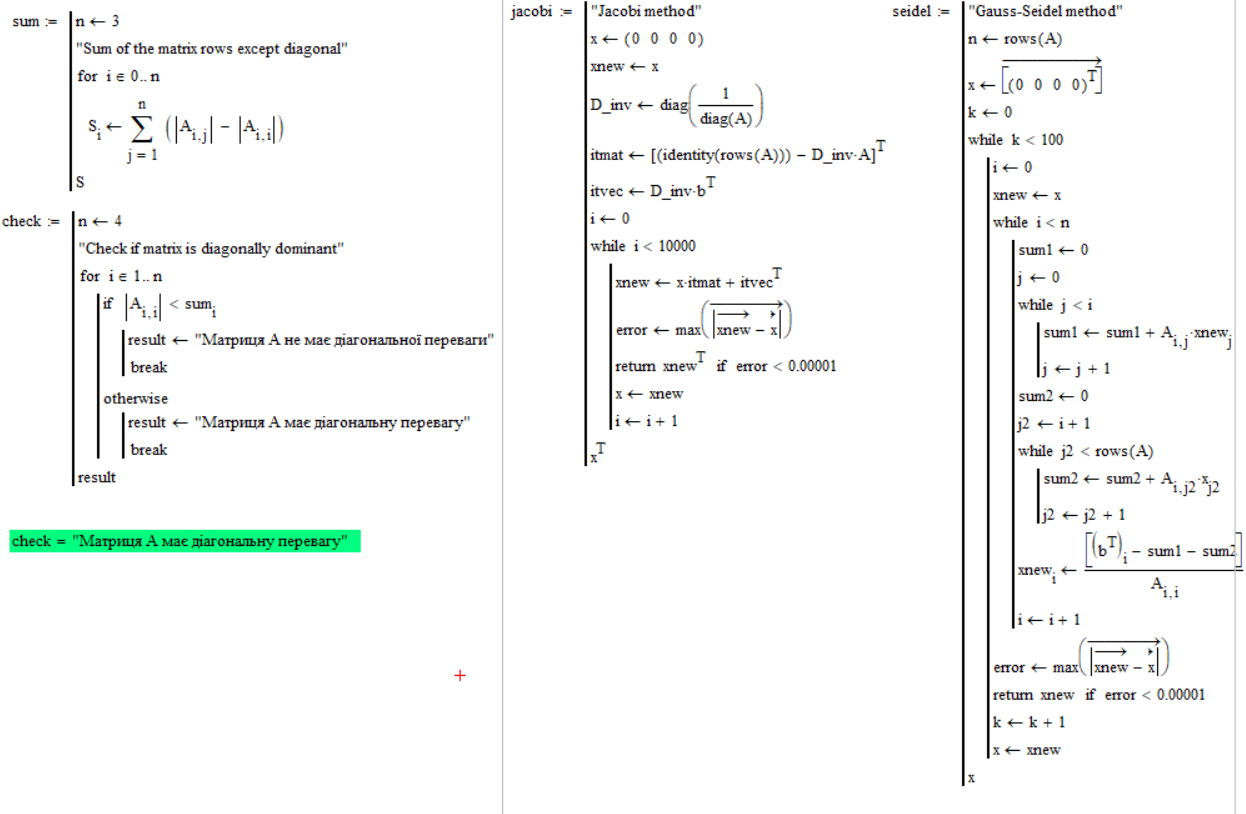
****

1. **Порівняння власного розв’язку та розв’язку отриманого у Mathcad**

****

****

1. **Лістинг розв’язку Mathcad**

****

1. **Лістинг програми**

| import numpy as np  from lib\_print import \*  np.set\_printoptions(precision=6, suppress=True, floatmode="fixed")  def is\_strictly\_diagonally\_dominant(matrix):  for i in range(matrix.shape[0]):  if abs(matrix[i, i]) <= np.sum(np.abs(matrix[i, :])) - abs(matrix[i, i]):  return False  return True  # за завданням  vector\_b = np.array([11.172, 0.115, 0.009, 9.349]).T  matrix\_a = np.array([  [2.12, 0.42, 1.34, 0.88],  [0.42, 3.95, 1.87, 0.43],  [1.34, 1.87, 2.98, 0.46],  [0.88, 0.43, 0.46, 4.44],  ])  # результат допрограмового етапу  matrix\_a = np.array([  [1.787, -0.758, -0.277, 0.127],  [0.42, 3.95, 1.87, 0.43],  [0.28, 1.66, 2.31, 0.02],  [0.88, 0.43, 0.46, 4.44]])  vector\_b = np.array([13.332, 0.115, -5.577, 9.349])    print(is\_strictly\_diagonally\_dominant(matrix\_a))    a = matrix\_a.copy()  b = vector\_b.copy()  n = a.shape[0]  print(printer(a, f"{ANSI.Styles.BOLD} Matrix A{ANSI.Styles.RESET}", default\_style=ANSI.Styles.ITALIC), end="\n\n")  print(printer(b.reshape(-1, 1), f"{ANSI.Styles.BOLD} Vector b{ANSI.Styles.RESET}", default\_style=ANSI.Styles.ITALIC), end="\n\n")  def jacobi(a, b, epsilon=1e-5, max\_iterations=1000):  \_\_print\_strouput = f" {ANSI.Styles.BOLD}{ANSI.FG.BLUE}JACOBI ALGORYTHM{ANSI.Styles.RESET}\n\n"  \_\_default\_print\_style = ANSI.FG.BRIGHT\_BLACK  n = a.shape[0]  D\_inv = np.diag(1 / np.diag(a))  iteration\_matrix = np.eye(n) - D\_inv @ a  iteration\_vector = D\_inv @ b  x = np.zeros\_like(b)      \_\_print\_highlights\_diag = [highlight([(i, i) for i in range(a.shape[0])], ANSI.FG.BLUE, 1, f"{ANSI.Styles.BOLD}Diagonal ")]  \_\_print\_highlights\_not\_diag = [highlight([(i, i) for i in range(a.shape[0])], ANSI.FG.BRIGHT\_BLACK, 1, f"{ANSI.Styles.BOLD}Diagonal ")]  \_\_print\_strouput += printer(D\_inv, ANSI.Styles.BOLD +" (jacobi) D inverted", \_\_print\_highlights\_diag, default\_style=ANSI.Styles.ITALIC+\_\_default\_print\_style, formatting="+0.5f") + "\n\n"  \_\_print\_strouput += printer(iteration\_matrix, ANSI.Styles.BOLD +" (jacobi) iteration matrix ", \_\_print\_highlights\_not\_diag, formatting="+0.5f", default\_style=ANSI.Styles.ITALIC) + "\n\n"  \_\_print\_strouput += printer(iteration\_vector.reshape(-1, 1), ANSI.Styles.BOLD +" (jacobi) iteration vector ", default\_style=ANSI.Styles.ITALIC) + '\n\n'  for i in range(max\_iterations):  x\_new = iteration\_matrix @ x + iteration\_vector  error = np.max(np.abs(x\_new - x))  \_\_print\_strouput += "\n" + printer((a @ x\_new - b).reshape(-1, 1), f"\n\n{ANSI.Styles.BOLD} (jacobi) a @ x - b; iteration {1+i:2d} {ANSI.Styles.RESET}", formatting= "+0.5f", default\_style=ANSI.FG.BRIGHT\_BLACK+ANSI.Styles.ITALIC, pre\_row\_str=" ") + "\n"  \_\_print\_strouput += printer(x\_new.reshape(-1, 1), f"{ANSI.Styles.BOLD} -> x, with change (x\_new - x) {ANSI.FG.RED} {error:0.5f} ", [], "+0.5f", default\_style=ANSI.Styles.ITALIC+ANSI.FG.BRIGHT\_BLACK, pre\_row\_str=" ")  if error < epsilon:    \_\_print\_strouput += printer((a @ x\_new - b).reshape(-1, 1), f"\n\n{ANSI.Styles.BOLD} (jacobi) a @ x - b, change (x\_new - x): {ANSI.FG.RED} {error:0.5f}{ANSI.Styles.RESET}", default\_style=ANSI.FG.GREEN+ANSI.Styles.ITALIC, formatting="0.5f")    print(\_\_print\_strouput)    return x\_new  x = x\_new    return x  def seidel(a, b, epsilon=1e-5, max\_iterations=1000):  n = a.shape[0]  x = np.zeros\_like(b)    \_\_print\_strouput = f" {ANSI.Styles.BOLD}{ANSI.FG.BLUE}Seidel ALGORYTHM{ANSI.Styles.RESET}"      for k in range(max\_iterations):  x\_new = np.zeros\_like(x)    for i in range(n):  sum1 = sum(a[i,j] \* x\_new[j] for j in range(i))  sum2 = sum(a[i,j] \* x[j] for j in range(i+1, n))  x\_new[i] = (b[i] - sum1 - sum2) / a[i,i]  error = np.max(np.abs(x\_new - x))  \_\_print\_strouput += "\n" + printer((a @ x\_new - b).reshape(-1, 1), f"\n\n{ANSI.Styles.BOLD} (seidel) a @ x - b; iteration {1+k:2d} {ANSI.Styles.RESET}", formatting= "+0.5f", default\_style=ANSI.FG.BRIGHT\_BLACK+ANSI.Styles.ITALIC, pre\_row\_str=" ") + "\n"  \_\_print\_strouput += printer(x\_new.reshape(-1, 1), f"{ANSI.Styles.BOLD} -> x, with change (x\_new - x) {ANSI.FG.RED} {error:0.5f} ", [], "+0.5f", default\_style=ANSI.FG.BRIGHT\_BLACK+ANSI.Styles.ITALIC, pre\_row\_str=" ")      if error < epsilon:  break  x = x\_new  \_\_print\_strouput += printer((a @ x\_new - b).reshape(-1, 1), f"{ANSI.Styles.BOLD}\n\n (seidel) a @ x - b, change (x\_new - x): {ANSI.FG.RED} {error:0.5f}{ANSI.Styles.RESET}, {ANSI.FG.BLUE}epsilon: {epsilon:0.5f}", default\_style=ANSI.Styles.ITALIC+ANSI.FG.GREEN, formatting="0.5f")  print(\_\_print\_strouput)  return x  sei = seidel(a.copy(), b.copy())  print()  jac = jacobi(a.copy(), b.copy())  print(printer((jac).reshape(-1, 1), f"{ANSI.Styles.BOLD}\n\n (jacobi) {ANSI.FG.RED}{ANSI.Styles.RESET}", default\_style=ANSI.Styles.ITALIC+ANSI.FG.GREEN, formatting="0.5f"), end="")  print(printer((sei).reshape(-1, 1), f"{ANSI.Styles.BOLD}\n\n (seidel) {ANSI.FG.RED}{ANSI.Styles.RESET}", default\_style=ANSI.Styles.ITALIC+ANSI.FG.GREEN, formatting="0.5f")) |
| --- |