1. **ОПИС АЛГОРИТМІВ**

Перелік всіх основних змінних та їхнє призначення наведено в

таблиці 3.1.

Таблиця 3.1 – Основні змінні та їхні призначення

|  |  |
| --- | --- |
| Змінна | Призначення |
| Matrix | Розширена матриця системи |
| result | Масив знайдений значень змінних системи |
| permittingElement | Головний елемент(у методі Жордана-Гауса) |
| *c, s* | Допоміжні елементи в методі обертань |

## Загальний алгоритм

1. ПОЧАТОК ;
2. Зчитати розмірність системи;
3. Зчитати матрицю системи та стовпець вільних членів:
   1. Зчитати матрицю системи:
      1. Цикл проходу по всіх рядках матриці системи (:
         1. Цикл проходу всіх стовпцях матриці системи   
            (:
            1. ЯКЩО поточний елемент матриці – вірно записане число, ТО записати його в відповідну комірку *matrix*. ІНАКШЕ видати повідомлення про помилку та перейти до пункту 8;
   2. Зчитати стовпець вільних членів:
      1. Цикл проходу по всіх елементах стовпця вільного членів:
         1. Якщо поточний елемент стовпцю вільних членів – вірно записане число, ТО записати його в відповідну комірку *free*. ІНАКШЕ видати повідомлення про помилку та перейти до пункту 8;
4. ЯКЩО обраний метод Гауса, ТО обробити дані згідно алгоритму методу Гауса (пункт 3.2);
5. ЯКЩО обраний метод Жордана-Гауса, ТО обробити дані згідно алгоритму методу Жордана-Гауса (пункт 3.3);
6. ЯКЩО обраний метод обертань, ТО обробити дані згідно алгоритму методу обертань(пункт 3.4);
7. ЯКЩО обраний метод сходиться для вхідних даних, ТО:
   1. ЯКЩО обрана система на дві невідомих, ТО побудувати та вивести графік системи;
   2. Вивести рішення системи;
   3. Записати систему та її рішення у файл;
8. КІНЕЦЬ.

## Алгоритм методу Гауса

1. ПОЧАТОК;
2. Шляхом елементарних перетворень знайти ранг розширеної та нерозширеної матриць:
   1. ЯКЩО ранг розширеної матриці більший за ранг основної ТО:

Вивести повідомлення : «Матриця несумісна, розв’язків не має»;

Перейти до пункту 7;

ІНАКШЕ ЯКЩО ранг розширеної матриці менший за ранг основної ТО :

Вивести повідомлення : «Матриця сумісна і має безліч розв’язків», перейти до пункту 7;

ІНАКШЕ

Продовжити;

1. Приведення матриці до канонічного (діагонального) вигляду (**stableMatrix**):
   * 1. Поділити елементи першого рядку на його перший елемент
     2. ЦИКЛ проходу по всіх рядках матриці (*Matrix*):
        1. ЦИКЛ проходу по всіх елементах поточного рядка у матриці (*Matrix*):
        2. Відняти відповідні елементи першого рядка, домножені на значення першого елемента в поточному рядку від елементів останнього;
2. Зворотній хід:
   1. Цикл проходу по всіх рядках матриці(*Matrix*), починаючи з останнього:
      1. Розділити поточний рядок на перше ненульове значення;
      2. Отримане значення в стовбці вільних членів помножити на усі елементи наступного по циклу рядка, крім першого ненульового;
3. Ініціювати масив чисел (*result*) довжиною кількості рядків матриці(*Matrix*);
4. Цикл проходу по індексам масиву (*result*) :
   1. Присвоїти значення відповідного елемента по індексу стовпця вільних членів елементу масиву (*result*);
5. Кінець.

Метод Жордана-Гауса :

1. ПОЧАТОК;
2. Шляхом елементарних перетворень знайти ранг розширеної та нерозширеної матриць:
   1. ЯКЩО ранг розширеної матриці більший за ранг основної ТО:

Вивести повідомлення : «Матриця несумісна, розв’язків не має»,

Перейти до пункту 4;

ІНАКШЕ ЯКЩО ранг розширеної матриці менший за ранг основної ТО :

Вивести повідомлення : «Матриця сумісна і має безліч розв’язків», перейти до пункту 4;

ІНАКШЕ:

Продовжити;

1. Приведеня основної матриці до одиничного:
   1. Цикл(1) проходу по всіх рядках матриці(*Matrix*):
      1. Ініціювати чисельну змінну (*permittingElement*) ;
      2. Надати їй значення елементу поточного рядка з індексом стовпця поточного рядка(елемент головної діагоналі) ;
      3. Цикл проходу по всіх елементах поточного рядка матриці(*Matrix*):
         1. Розділити елемент поточного індекса по стовбцю на (*permittingElement*);
      4. Надати змінній (*permittingElement*) значення елементу головної діагоналі поточного рядка;
      5. Цикл проходу по елементах рядка заданої матриці (*Matrix*)(починаючи з індексу, на один більший за поточний індекс рядка):
         1. Цикл(2) походу по стовбцях матриці (*Matrix*):
            1. Якщо індекс циклу(2) не дорівнює індексу циклу(1);

Помножити (*permittingElement*) на елемент матриці (*Matrix*)по індексам :[поточний індекс циклу по елементах рядка][поточний індекс циклу(2)] та відняти від отриманого результату добуток елементів по індексам :[поточний індекс циклу(1)][ поточний індекс циклу по елементах рядка], [поточний індекс циклу(2)][ поточний індекс циклу(1)];

Елементу матриці (*Matrix*) по індексам: [поточний індекс циклу по елементах рядка][почтовий індекс циклу(2)] присвоїти отримане значення;

* + 1. Онулювати значення ствобця поточного індекса циклу(1) крім елементу головної діагоналі;
    2. Ініціювати чисельний масив(*result*) довжиною кількості змінних системи;
    3. Цикл проходу по масиву(*result*):
       1. Присвоїти значення відповідного елемента по індексу стовпця вільних членів елементу масиву (*result*);

1. КІНЕЦЬ.

Метод обертань:

1. ПОЧАТОК
2. Шляхом елементарних перетворень знайти ранг розширеної та нерозширеної матриць:

2.1 ЯКЩО ранг розширеної матриці більший за ранг основної ТО:

Вивести повідомлення : «Матриця несумісна, розв’язків не має»,

Перейти до пункту 4;

ІНАКШЕ ЯКЩО ранг розширеної матриці менший за ранг основної ТО :

Вивести повідомлення : «Матриця сумісна і має безліч розв’язків», перейти до пункту 4;

ІНАКШЕ:

Продовжити;

1. Ітерація(1) по рівняннях матриці(matrix):

3.1 Ітерація(2) по рівняннях матриці(matrix), починаючи з рівняння, на один нижче, від поточного рівняння ітерації(1);

3.1.1 Надати змінним *c,s* значень коефіцієнта головної діагоналі матриці поточного рівняння ітерації(1) та коефіцієнта поточного рівняння ітерації(2), що знаходиться під першим відповідно;

3.1.2 Ітерація(3) по стовпцях матриці;

3.1.2.1 Надати коефіцієнту поточного стовпця ітерації(3) поточного рівняння ітерації(1) значення суми добутків коефіцієнта *с*  на цей самий елемент та коефіцієнта *s* на коефіцієнт поточного стовпця ітерації(2) поточного рівняння ітерації(1);

3.1.2.2 Надати коефіцієнту поточного стовпця ітерації(3) поточного рівняння ітерації(2) значення різниці добутків коефіцієнта *s*  на цей самий елемент та коефіцієнта *с* на коефіцієнт поточного стовпця ітерації(3) поточного рівняння ітерації(1)(до надання йому нового значення);

1. Зворотній хід:
   1. Цикл проходу по всіх рядках матриці(*Matrix*), починаючи з останнього:

4.1.1Розділити поточний рядок на перше ненульове значення;

4.1.2 Отримане значення в стовбці вільних членів помножити на усі елементи наступного по циклу рядка, крім першого ненульового;

1. Ініціювати масив чисел (*result*) довжиною кількості змінних матриці(*Matrix*);
2. Цикл проходу по індексам масиву (*result*) :
   1. Присвоїти значення відповідного елемента по індексу стовпця вільних членів елементу масиву (*result*);
3. Кінець.