

Лабораторная работа 4. Системы линейных уравнений

Отчет по лабораторной работе 4

Милёхин Александр НПИМд-02-21

Содержание

1	Цель работы	4
2	Теоретические сведения	5
3	Задание	6
4	Выполнение лабораторной работы	7
5	Выводы	18

List of Figures

4.1	Расширенная матрица	7
4.2	Элемент матрицы	8
4.3	Вектор строки	9
4.4	Преобразование матрицы. Шаг 1	10
4.5	Преобразование матрицы. Шаг 2	11
4.6	Получение единичной матрицы	12
4.7	Более высокая точность записи десятичного числа	12
4.8	Короткая форма записи десятичного числа	13
4.9	Выделение матрицы и вектора	14
4.10	Вектор x	15
4.11	Матрица A	16
4.12	LU-разложение матрицы A	17

1 Цель работы

Познакомиться с методами исследования систем линейных уравнений в Octave.

2 Теоретические сведения

Вся теоретическая часть по выполнению лабораторной работы была взята из инструкции по лабораторной работе №4 (“Лабораторная работа №4. Описание”) на сайте: <https://esystem.rudn.ru/course/view.php?id=12766>

3 Задание

Выполните работу и задокументируйте процесс выполнения.

4 Выполнение лабораторной работы

1. Метод Гаусса

Octave содержит сложные алгоритмы, встроенные для решения систем линейных уравнений.

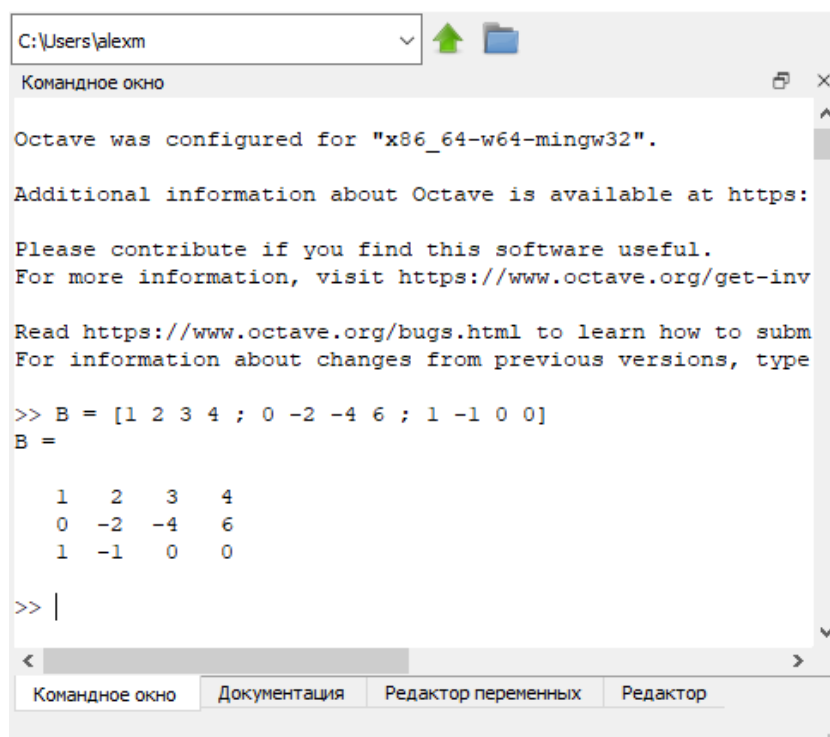
Для решения системы линейных уравнений:

$$Ax = b$$

методом Гаусса можно построить расширенную матрицу вида:

$$B = (A|b).$$

Рассмотрим расширенную матрицу.



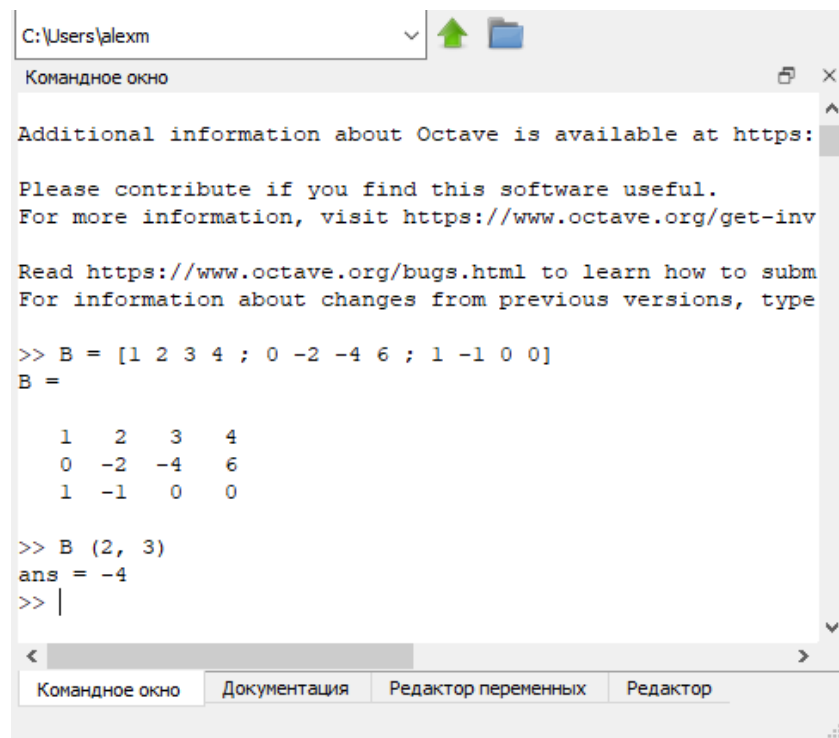
The screenshot shows the Octave Command Window interface. At the top, the path 'C:\Users\alexm' is displayed. Below it, the title bar reads 'Командное окно'. The main area contains the following text:

```
Octave was configured for "x86_64-w64-mingw32".  
Additional information about Octave is available at https:  
Please contribute if you find this software useful.  
For more information, visit https://www.octave.org/get-inv  
Read https://www.octave.org/bugs.html to learn how to subm  
For information about changes from previous versions, type  
  
>> B = [1 2 3 4 ; 0 -2 -4 6 ; 1 -1 0 0]  
B =  
  
    1    2    3    4  
    0   -2   -4    6  
    1   -1    0    0  
  
>> |
```

At the bottom, there are four tabs: 'Командное окно' (selected), 'Документация', 'Редактор переменных', and 'Редактор'.

Figure 4.1: Расширенная матрица

Ее можно просматривать поэлементно.



```
C:\Users\alexm
Командное окно

Additional information about Octave is available at https:
Please contribute if you find this software useful.
For more information, visit https://www.octave.org/get-inv
Read https://www.octave.org/bugs.html to learn how to subm
For information about changes from previous versions, type

>> B = [1 2 3 4 ; 0 -2 -4 6 ; 1 -1 0 0]
B =

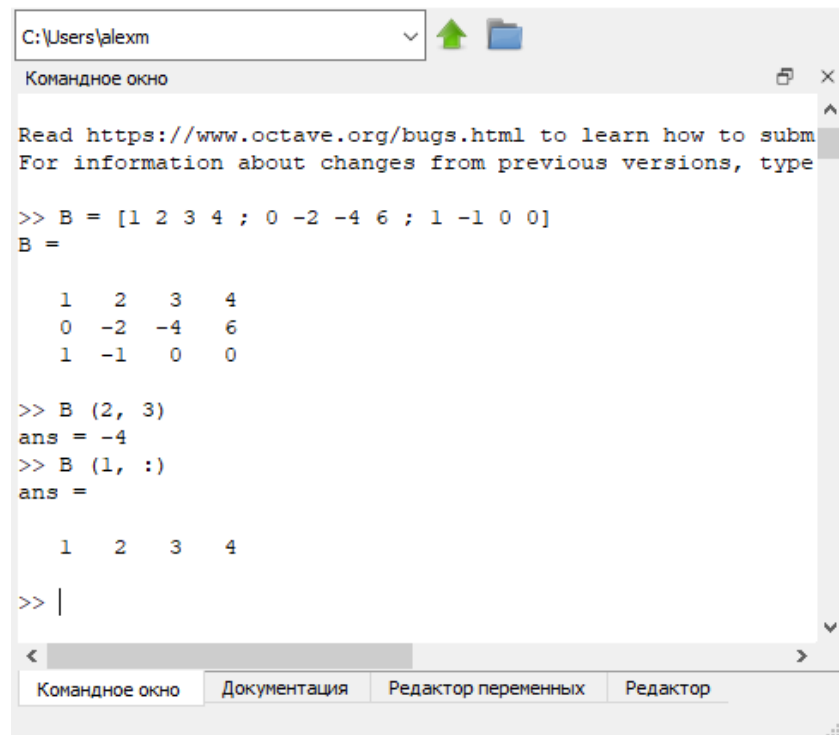
    1    2    3    4
    0   -2   -4    6
    1   -1    0    0

>> B (2, 3)
ans = -4
>> |
```

Figure 4.2: Элемент матрицы

Это скаляр, хранящийся в строке 2, столбце 3.

Также можно извлечь целый вектор строки или вектор столбца, используя оператор сечения. Сечение можно использовать для указания ограниченного диапазона. Если не указано начальное или конечное значение, то результатом оператора является полный диапазон (см. Fig. 3).



The screenshot shows the Octave Command Window interface. At the top, there is a path bar showing 'C:\Users\alexm' and a toolbar with icons for home, up, and down. Below the path bar, the window title is 'Командное окно'. The main text area contains the following text: 'Read <https://www.octave.org/bugs.html> to learn how to submit bugs. For information about changes from previous versions, type `info`'. The user has entered the command `>> B = [1 2 3 4 ; 0 -2 -4 6 ; 1 -1 0 0]`, and the output shows the matrix `B =` as a 3x4 matrix: $\begin{bmatrix} 1 & 2 & 3 & 4 \\ 0 & -2 & -4 & 6 \\ 1 & -1 & 0 & 0 \end{bmatrix}$. The user then enters `>> B (2, 3)`, and the output is `ans = -4`. Next, the user enters `>> B (1, :)`, and the output is `ans =` followed by the first row of the matrix: $\begin{bmatrix} 1 & 2 & 3 & 4 \end{bmatrix}$. The window has a scrollbar on the right and a tab bar at the bottom with tabs for 'Командное окно', 'Документация', 'Редактор переменных', and 'Редактор'.

```
C:\Users\alexm
Командное окно

Read https://www.octave.org/bugs.html to learn how to submit bugs.
For information about changes from previous versions, type info

>> B = [1 2 3 4 ; 0 -2 -4 6 ; 1 -1 0 0]
B =

    1    2    3    4
    0   -2   -4    6
    1   -1    0    0

>> B (2, 3)
ans = -4

>> B (1, :)
ans =

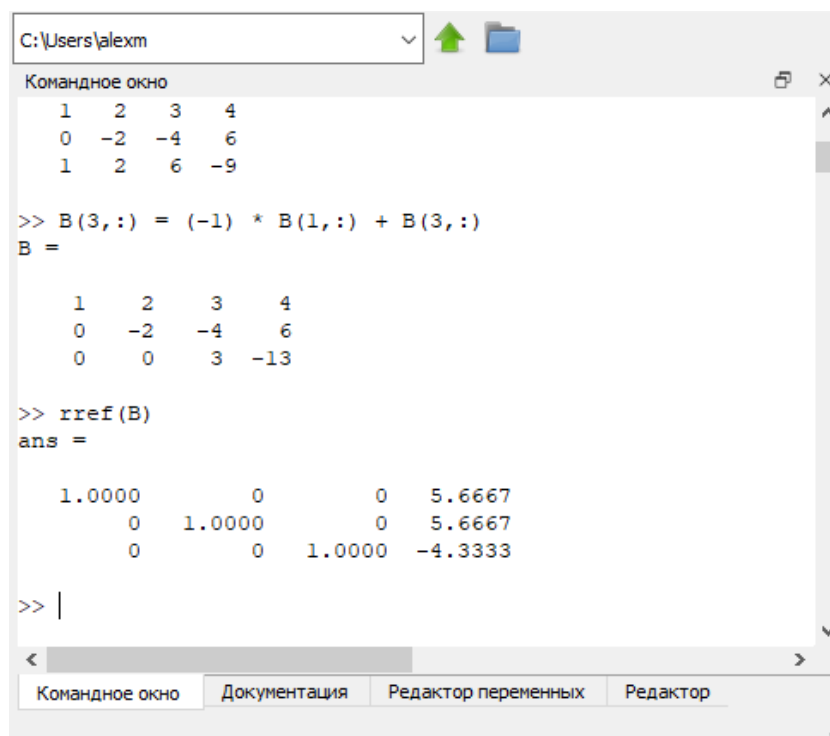
    1    2    3    4

>> |
```

Figure 4.3: Вектор строки

Реализуем теперь явно метод Гаусса.

Сначала добавим к третьей строке первую строку, умноженную на -1 .



The image shows a MATLAB Command Window with the following content:

```
C:\Users\alexm
Командное окно
1  2  3  4
0 -2 -4  6
1  2  6 -9

>> B(3,:) = (-1) * B(1,:) + B(3,:)
B =

    1     2     3     4
    0    -2    -4     6
    0     0     3   -13

>> rref(B)
ans =

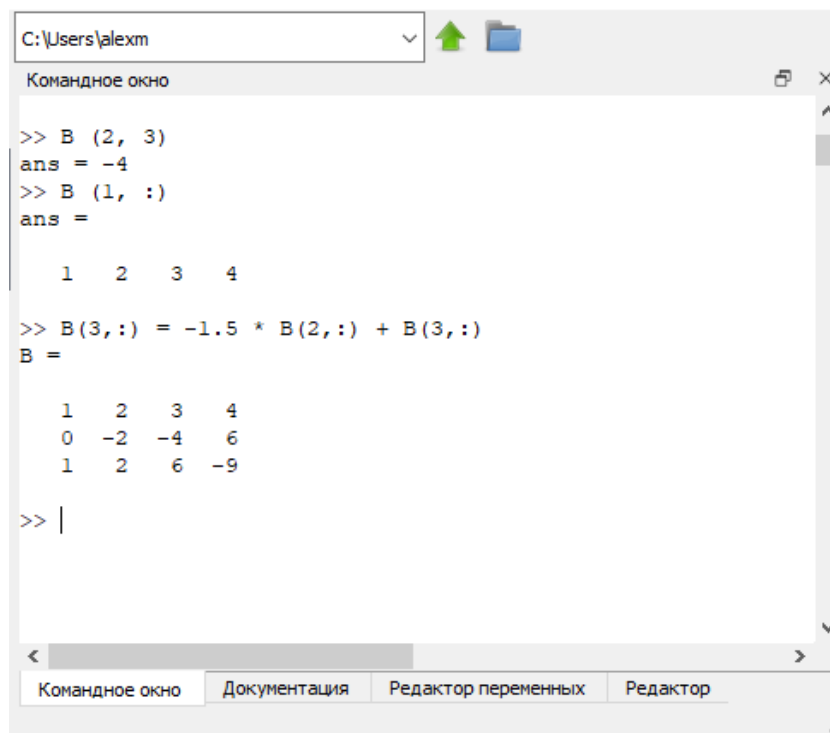
    1.0000     0     0    5.6667
     0    1.0000     0    5.6667
     0     0    1.0000   -4.3333

>> |
```

At the bottom, there are tabs for 'Командное окно', 'Документация', 'Редактор переменных', and 'Редактор'.

Figure 4.4: Преобразование матрицы. Шаг 1

Далее добавим к третьей строке вторую строку, умноженную на -1.5 .



The screenshot shows the Octave Command Window with the following commands and outputs:

```
>> B (2, 3)
ans = -4
>> B (1, :)
ans =
    1    2    3    4
>> B(3,:) = -1.5 * B(2,:) + B(3,:)
B =
    1    2    3    4
    0   -2   -4    6
    1    2    6   -9
>> |
```

The window title is "Командное окно" and the path is "C:\Users\alexm". The bottom bar shows tabs for "Командное окно", "Документация", "Редактор переменных", and "Редактор".

Figure 4.5: Преобразование матрицы. Шаг 2

Матрица теперь имеет треугольный вид. Очевидным образом получим ответ: 5.66667; 5.66667; -4.33333

Этот ответ был получен путем решения третьей строки матрицы, а впоследствии подставлением найденных элементов в другие строки матрицы. Либо этот ответ можно получить приведя матрицу к единичной (треугольной), цифры справа — это и есть ответ.

Конечно, Octave располагает встроенной командой для непосредственного поиска треугольной формы матрицы.

```
C:\Users\alexm
Командное окно
0 -2 -4 6
0 0 3 -13

>> rref(B)
ans =

1.0000 0 0 5.6667
0 1.0000 0 5.6667
0 0 1.0000 -4.3333

>> format long
>> rref(B)
ans =

1.0000000000000000 0 0 5.666666666666667
0 1.0000000000000000 0 5.666666666666666
0 0 1.0000000000000000 -4.333333333333333

>> |
```

Figure 4.6: Получение единичной матрицы

Следует обратить внимание, что все числа записываются в виде чисел с плавающей точкой (то есть десятичных дробей). Пять десятичных знаков отображаются по умолчанию. Переменные на самом деле хранятся с более высокой точностью, и при желании можно отобразить больше десятичных разрядов.

```
C:\Users\alexm
Командное окно
0 -2 -4 6
0 0 3 -13

>> rref(B)
ans =

1.0000 0 0 5.6667
0 1.0000 0 5.6667
0 0 1.0000 -4.3333

>> format long
>> rref(B)
ans =

1.0000000000000000 0 0 5.666666666666667
0 1.0000000000000000 0 5.666666666666666
0 0 1.0000000000000000 -4.333333333333333

>> |
```

Figure 4.7: Более высокая точность записи десятичного числа

Вернем предыдущий формат представления.

```

C:\Users\alexm
Командное окно

1  2  3  4
0 -2 -4  6
1  2  6 -9

>> B(3,:) = (-1) * B(1,:) + B(3,:)
B =

1  2  3  4
0 -2 -4  6
0  0  3 -13

>> rref(B)
ans =

1.0000    0    0  5.6667
0  1.0000    0  5.6667
0    0  1.0000 -4.3333

>> format long
>> rref(B)
ans =

1.0000000000000000    0    0  5.6666666666
0  1.0000000000000000    0  5.6666666666
0    0  1.0000000000000000 -4.3333333333

>> format short
>> |

```

Figure 4.8: Короткая форма записи десятичного числа

2. Левое деление

Встроенная операция для решения линейных систем вида

$$Ax = b$$

в Octave называется левым делением и записывается как $A \setminus b$. Выделим из расширенной матрицы B матрицу A , а также вектор b .

```
C:\Users\alexm
Командное окно
>> rref(B)
ans =

    1.0000000000000000    0    0    5.6666666666
         0    1.0000000000000000    0    5.6666666666
         0    0    1.0000000000000000   -4.3333333333

>> format short
>> B = [1 2 3 4 ; 0 -2 -4 6 ; 1 -1 0 0]
B =

     1     2     3     4
     0    -2    -4     6
     1    -1     0     0

>> A = B(:,1:3)
A =

     1     2     3
     0    -2    -4
     1    -1     0

>> b = B(:,4)
b =

     4
     6
     0

>> |
```

Figure 4.9: Выделение матрицы и вектора

После чего найдём вектор x .

```
C:\Users\alexm
Командное окно
>> format short
>> B = [1 2 3 4 ; 0 -2 -4 6 ; 1 -1 0 0]
B =

     1     2     3     4
     0    -2    -4     6
     1    -1     0     0

>> A = B(:,1:3)
A =

     1     2     3
     0    -2    -4
     1    -1     0

>> b = B(:,4)
b =

     4
     6
     0

>> A\b
ans =

    5.6667
    5.6667
   -4.3333

>> |
```

Figure 4.10: Вектор x

3. LU-разложение

- LU-разложение:

LU разложение – это вид факторизации матриц для метода Гаусса. Цель состоит в том, чтобы записать матрицу A в виде:

$$A = LU,$$

где L – нижняя треугольная матрица, а U – верхняя треугольная матрица. Эта факторизованная форма может быть использована для решения уравнения $Ax = b$.

LU-разложение существует только в том случае, когда матрица A обратима, а все главные миноры матрицы A невырождены. Этот метод является одной из разновидностей метода Гаусса.

- Решение систем линейных уравнений:

Если известно LU-разложение матрицы A , то исходная система может быть записана как:

$$LUx = b.$$

Эта система может быть решена в два шага. На первом шаге решается система:

$$Ly = b.$$

Поскольку L – нижняя треугольная матрица, эта система решается непосредственно прямой подстановкой.

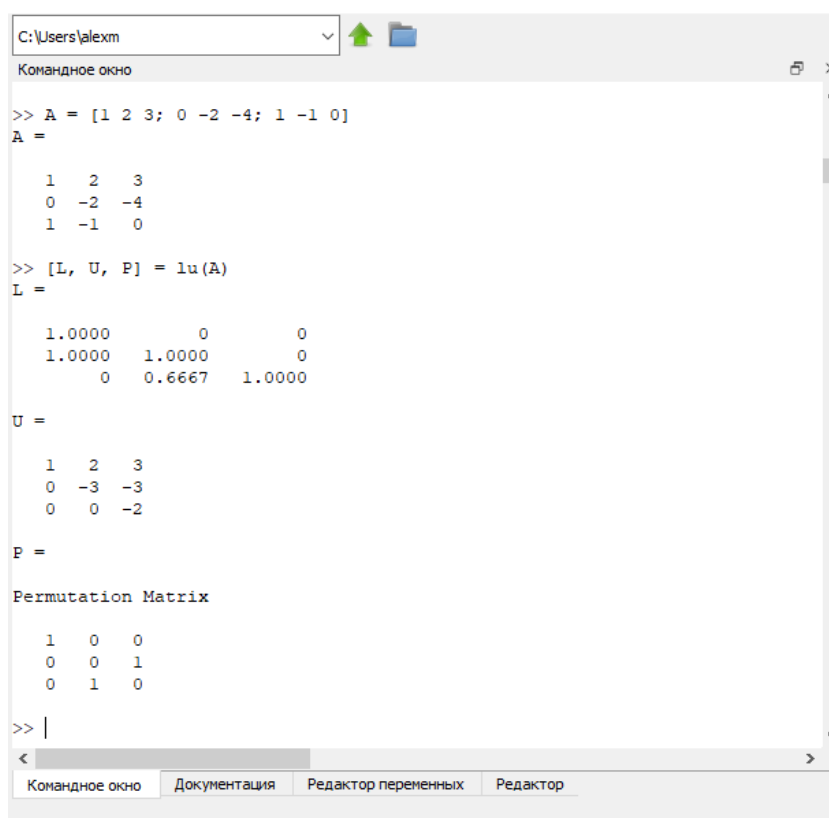
На втором шаге решается система:

$$Ux = y.$$

Поскольку U – верхняя треугольная матрица, эта система решается непосредственно обратной подстановкой.

- Задание:

Пусть дана матрица A .



```
>> A = [1 2 3; 0 -2 -4; 1 -1 0]
A =

     1     2     3
     0    -2    -4
     1    -1     0

>> [L, U, P] = lu(A)
L =

    1.0000         0         0
    1.0000    1.0000         0
         0    0.6667    1.0000

U =

     1     2     3
     0    -3    -3
     0     0    -2

P =

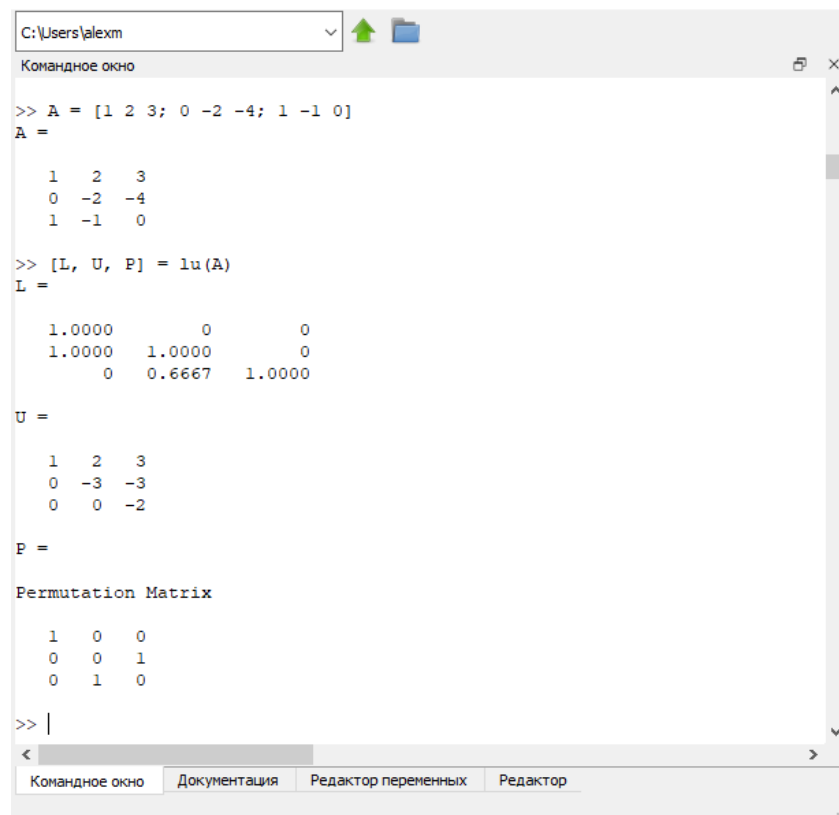
Permutation Matrix

     1     0     0
     0     0     1
     0     1     0

>> |
```

Figure 4.11: Матрица A

С помощью Octave нужно расписать её LU-разложение.
Распишем LU-разложение матрицы A.



```
C:\Users\alexm
Командное окно

>> A = [1 2 3; 0 -2 -4; 1 -1 0]
A =

     1     2     3
     0    -2    -4
     1    -1     0

>> [L, U, P] = lu(A)
L =

     1.0000         0         0
     1.0000     1.0000         0
           0     0.6667     1.0000

U =

     1     2     3
     0    -3    -3
     0     0    -2

P =

Permutation Matrix

     1     0     0
     0     0     1
     0     1     0

>> |
```

The screenshot shows the Octave Command Window with the following content: The title bar indicates the path 'C:\Users\alexm'. The window title is 'Командное окно'. The user has entered the command to create matrix A: `>> A = [1 2 3; 0 -2 -4; 1 -1 0]`. The output shows matrix A as a 3x3 matrix. Then, the user enters the command for LU decomposition: `>> [L, U, P] = lu(A)`. The output shows three matrices: L (lower triangular), U (upper triangular), and P (permutation matrix). Matrix L has values 1.0000, 1.0000, and 0.6667. Matrix U has values 1, -3, and -2. Matrix P is a permutation matrix with rows [1, 0, 0], [0, 0, 1], and [0, 1, 0]. The window has tabs for 'Командное окно', 'Документация', 'Редактор переменных', and 'Редактор'.

Figure 4.12: LU-разложение матрицы A

5 Выводы

Я познакомился с методами исследования систем линейных уравнений в Octave.