Министерство образования и науки Российской Федерации
Российский государственный университет нефти и газа
(национальный исследовательский университет)
имени И.М. Губкина

Факультет Автоматики и вычислительной техники
Кафедра Автоматизированных систем управления

Отчёт по лабораторной работе №1
«АНАЛИЗ ОБУСЛОВЛЕННОСТИ ЗАДАЧИ РЕШЕНИЯ СЛАУ»
По дисциплине «Вычислительные методы и математические пакеты»

Выполнил:
студент группы АС-23-04
Ханевский Ярослав

Проверила:
ст. преп. Степанкина О.А.

Москва, 2025 г.

Группа: АС-23-04. **ФИО:** Ханевский Ярослав. **Номер в списке:** 29.

Варианты данных: 4 (n = 8), 1 (первая норма - l_1).

Название работы: «Анализ обусловленности задачи решения СЛАУ».

1. Задать систему линейных уравнений по предложенным правилам:
   - вариант данных выбирается по номеру человека в списке группы, например, 1 вариант данных выбирают студенты с номерами 1, 6, 11, … Вариант размера системы n = 8, 6 или 7 выбирается по аналогичному правилу;
   - вариант формулировки заданий для этой работы один, но есть четыре варианта норм, с которыми следует согласовывать расчеты (вариант 1 - l_1, вариант 2 - l_2, вариант 3 - l_∞, вариант 4 - евклидова норма). Выбор осуществляется по номеру в списке, то есть $\|.\|_1$ используют студенты с номерами 1, 5, 9,…

Определим матрицу A размером n*n (n = 8 в варианте 4), изначально заполненную нулями (функция *zeros*). С помощью цикла for по заданным в варианте 4 задания правилам заполним эту матрицу:

Вариант 4.

$$A = \begin{bmatrix} y_1 & a_2 & a_3 & \dots & a_n \\ a_1 & y_2 & a_3 & \dots & a_n \\ a_1 & a_2 & y_3 & \dots & \dots \\ \dots & \dots & \dots & \dots & a_n \\ a_1 & a_2 & a_3 & \dots & y_n \end{bmatrix} \qquad b = \begin{bmatrix} n \\ 1 \\ 1 \\ \dots \\ 1 \end{bmatrix}$$

$$y_k = 1/k \qquad a_k = (-1)^k k$$

Рисунок 1. Вариант входных данных

```
clc;
clear;

n = 8;
A = zeros(n, n);
for i = 1:n
    for j = 1:n
        if (i == j)
            A(i, j) = 1/i;
        else
            A(i, j) = ((-1)^j)*j;
        end
    end
end
A
```

Рисунок 2. Код создания и отображения матрицы А

Это точная матрица коэффициентов системы:

```
A =

    1.0000    2.0000   -3.0000    4.0000   -5.0000    6.0000   -7.0000    8.0000
   -1.0000    0.5000   -3.0000    4.0000   -5.0000    6.0000   -7.0000    8.0000
   -1.0000    2.0000    0.3333    4.0000   -5.0000    6.0000   -7.0000    8.0000
   -1.0000    2.0000   -3.0000    0.2500   -5.0000    6.0000   -7.0000    8.0000
   -1.0000    2.0000   -3.0000    4.0000    0.2000    6.0000   -7.0000    8.0000
   -1.0000    2.0000   -3.0000    4.0000   -5.0000    0.1667   -7.0000    8.0000
   -1.0000    2.0000   -3.0000    4.0000   -5.0000    6.0000    0.1429    8.0000
   -1.0000    2.0000   -3.0000    4.0000   -5.0000    6.0000   -7.0000    0.1250
```

Рисунок 3. Точная матрица А

Зададим вектор B размером n*1, изначально заполненный единицами. Поменяем первый элемент вектора на n = 8:

```
B = ones(n, 1);
B(1, 1) = 8;
B
```

Рисунок 4. Код создания и отображения вектора B

Это точный вектор коэффициентов правой части системы (свободных членов):

```
B =

     8
     1
     1
     1
     1
     1
     1
     1
```

Рисунок 5. Точный вектор B

Найти решение системы.

Найдем решение СЛАУ, используя функцию *linsolve*(A, B), где A – матрица коэффициентов системы уравнений, B – вектор-столбец свободных членов:

```
21        X = linsolve(A, B)
```

Рисунок 6. Создание и отображения вектора X

```
X =

    3.1684
    0.4421
   -0.1989
    0.1768
   -0.1275
    0.1137
   -0.0928
    0.0842
```

Рисунок 7. Точный вектор X

Это будет точный вектор переменных.

2. Внести в матрицу коэффициентов и вектор свободных членов 5% шум. Для этого каждый элемент матрицы и вектора складывается с нормально распределённой случайной величиной с математическим ожиданием 0 и параметром разброса, обеспечивающим заданный уровень погрешности.

Найти возмущенное решение системы и погрешность решения для трех случаев:

- матрица коэффициентов задана точно, вектор свободных членов – приближенно;

- матрица коэффициентов задана приближенно, вектор свободных членов – точно;

- матрица коэффициентов и вектор свободных членов заданы приближенно.

Чтобы внести в матрицу коэффициентов и вектор свободных членов 5% шум, используем функцию *normrnd*(*mu*, *sigma*, *size*), позволяющую сгенерировать случайную матрицу чисел из нормального распределения, где *mu* – математическое ожидание (по условию равное 0), *sigma* – стандартное (среднеквадратичное) отклонение, *size* – размер матрицы:

```
noiseA = normrnd(0, 0.05/3, n, n);
noiseB = normrnd(0, 0.05/3, n, 1);
```

Рисунок 8. Создание матрицы и вектора шума

По условию задан 5% шум, который имеет нормальное распределение с математическим ожиданием 0, поэтому распределяется по *3*sigma*. В связи с этим установим вторым параметром функции 0.05/3.

Теперь каждый элемент матрицы A и вектора B складываем с полученными шумами:

```
An = A + noiseA
Bn = B + noiseB
```

Рисунок 9.Код создания и отображения возмущенных вектора B и матрицы A

Получаем следующие возмущенные матрицу An и вектор Bn – назовем их приближенными:

```
noiseA =

   -0.0032    0.0237    0.0036   -0.0014    0.0051    0.0226    0.0485    0.0117
    0.0148    0.0049   -0.0194   -0.0322   -0.0100   -0.0179    0.0138   -0.0342
   -0.0127    0.0033   -0.0191   -0.0073    0.0082    0.0160    0.0230   -0.0059
   -0.0234    0.0265    0.0017   -0.0299    0.0123    0.0021   -0.0176   -0.0137
   -0.0237   -0.0134    0.0120    0.0140    0.0285    0.0239   -0.0078   -0.0263
    0.0081    0.0116    0.0431   -0.0148   -0.0032   -0.0327   -0.0045    0.0085
   -0.0030    0.0139   -0.0111    0.0017   -0.0356   -0.0033    0.0183    0.0047
   -0.0033   -0.0041    0.0031   -0.0091   -0.0140   -0.0201   -0.0046    0.0006


noiseB =

   -0.0222
    0.0188
    0.0058
   -0.0050
    0.0004
   -0.0044
   -0.0292
   -0.0048
```

Рисунок 10. Возмущенные вектор B и матрица A

С помощью той же функции *linsolve* найдем возмущенные решение системы для трёх случаев:

```
Xn1 = linsolve(A, Bn)
Xn2 = linsolve(An, B)
Xn3 = linsolve(An, Bn)
```

Рисунок 11. Код создания и отображения возмущенных векторов X

```
Xn1 =

    3.1690
    0.4469
   -0.1926
    0.1778
   -0.1235
    0.1173
   -0.0929
    0.0859


Xn2 =

    3.1801
    0.4449
   -0.2228
    0.1653
   -0.1339
    0.1171
   -0.0925
    0.0762


Xn3 =

    3.1808
    0.4497
   -0.2163
    0.1663
   -0.1299
    0.1207
   -0.0926
    0.0779
```

Рисунок 12. Возмущенные векторы X

Относительную погрешность решения будем искать через нормы l_1 векторов X (точный вектор решения) и Xn (вектор решения с погрешностью) по следующему правилу:

$$\delta = \frac{\|\Delta x\|}{\|x_{\text{точное}}\|}; \text{ где } \Delta x = X - Xn, x_{\text{точное}} = X$$

Используем функцию $norm(x, p)$ для нахождения нормы вектора x, причем p может принимать значения 1 – норма l_1, 2 (или просто $norm(x)$) – норма l_2 и inf для поиска бесконечностной нормы. Тогда относительная погрешность, исходя из выше описанной формулы вычисляется следующим образом:

```
err1 = norm((X - Xn1), 1)/norm(X, 1)
err2 = norm((X - Xn2), 1)/norm(X, 1)
err3 = norm((X - Xn3), 1)/norm(X, 1)
```

Рисунок 13. Код вычисления и отображения относительных погрешностей векторов X

```
err1 =

    0.0087


err2 =

    0.0320


err3 =

    0.0391
```

Рисунок 14. Относительные погрешности векторов X

Вычислить δрешения/δданных.

Для нахождения относительных погрешностей входных данных воспользуемся формулой:

$$\delta_{\text{вх}} = \max(\delta_A, \delta_B)\,;\ \text{где}\ \delta_A = \frac{\|A - An\|}{\|A_{\text{точное}}\|}, \delta_B = \frac{\|B - Bn\|}{\|B_{\text{точное}}\|}$$

```
errInp1 = norm((B - Bn), 1)/norm(B, 1);
errInp2 = norm((A - An), 1)/norm(A, 1);
errInp3 = max(norm((A - An), 1)/norm(A, 1), norm((B - Bn), 1)/norm(B, 1));
dd1 = err1/errInp1
dd2 = err2/errInp2
dd3 = err3/errInp3
```

Рисунок 15. Код вычисления относительных погрешностей входных данных и □решения/□данных

Значения δрешения/δданных:

```
dd1 =

    0.8360


dd2 =

    7.1797


dd3 =

    2.7621
```

Рисунок 16. ⍰решения/⍰данных

3. Рассчитать числа обусловленности для исходной и возмущенной системы. Оценить верхнюю границу относительной погрешности для каждого случая.

Число обусловленности ищется через произведение норм данной матрицы А и обратной ей матрицы:

$$\|A\| \cdot \|A^{-1}\|$$

От этой величины зависит степень влияния погрешности коэффициентов системы уравнений (матрицы А) на погрешность полученного решения (вектора X). Чем больше число обусловленности, тем больше будет влияние погрешности коэффициентов на погрешность решения.

В matlab для поиска числа обусловленности имеется встроенная функция *cond*(*A*, *p*), где *A* – матрица, *p* может принимать значения *1*, *2*, *inf* или *'fro'* в зависимости от используемой нормы. Используем заданную в варианте первую норму (*p* = 1).

Для точной матрицы А:

```
condA = cond(A, 1)
```

Рисунок 17. Код вычисления и отображения числа обусловленности для матрицы А

```
condA =

   48.7831
```

Рисунок 18. Число обусловленности для матрицы А

Для приближенной(возмущенной) матрицы А:

```
condAn = cond(An, 1)
```

Рисунок 19. Код вычисления и отображения числа обусловленности для приближенной матрицы А

```
condAn =

   48.0010
```

Рисунок 20. Число обусловленности для приближенной матрицы А

Чтобы рассчитать естественное число обусловленности(ЕЧО), воспользуемся формулой:

$$\|A^{-1}\| \cdot \|b\| / \|x\|$$

Оно зависит от конкретного решения x и характеризует коэффициент возможного возрастания относительной погрешности этого решения, вызванного погрешностью входных данных.

Для точной матрицы А:

```
echoA = norm(inv(A), 1)*norm(B, 1)/norm(X, 1)
```

Рисунок 21. Код вычисления и отображения ЕЧО для матрицы А

```
echoA =

    2.9601
```

Рисунок 22. ЕЧО для матрицы А

Для трёх случаев возмущенных входных данных:

```
echoAn1 = norm(inv(A), 1)*norm(Bn, 1)/norm(Xn1, 1)
echoAn2 = norm(inv(An), 1)*norm(B, 1)/norm(Xn2, 1)
echoAn3 = norm(inv(An), 1)*norm(Bn, 1)/norm(Xn3, 1)
```

Рисунок 23. Код вычисления и отображения ЕЧО для трёх случаев возмущенных входных данных

```
echoAn1 =

    2.9463


echoAn2 =

    3.0043


echoAn3 =

    2.9903
```

Рисунок 24. ЕЧО для трёх случаев возмущенных входных данных

Для оценки относительной погрешности используем следующие формулы:

$$\frac{\|\Delta X\|}{\|X\|} \leq \|A\| * \|A^{-1}\| * \frac{\|\Delta B\|}{\|B\|}$$

$$\frac{\|\Delta X\|}{\|X\|} \leq \|A\| * \|A^{-1}\| * \frac{\|\Delta A\|}{\|A\|}$$

Первое неравенство справедливо, когда рассматриваем случай с заданной точно матрицей A и заданным приближенно вектором B. Второе – когда матрица A задана приближенно, а вектор B – точно. Также когда матрица A задана точно, а вектор B приближенно, можно пользоваться следующей формулой:

$$\frac{\|\Delta X\|}{\|X\|} \leq nature * \frac{\|\Delta B\|}{\|B\|}$$

Когда и матрица A задана приближенно и вектор B приближенно, применяется следующая формула:

$$\frac{\|\Delta X\|}{\|X\|} \leq \|A\| * \|A^{-1}\| * \max\left(\frac{\|\Delta A\|}{\|A\|}, \frac{\|\Delta B\|}{\|B\|}\right)$$

Для трёх случаев возмущенных входных данных:

```
fprintf(1, '%g <= %g \n', err1, condAn*errInp1)
fprintf(1, '%g <= %g \n', err2, condAn*errInp2)
fprintf(1, '%g <= %g \n', err3, condAn*errInp3)
```

Рисунок 25. Код для вывода неравенств для трёх случаев возмущенных входных данных

```
0.00564372 <= 0.273763
0.0252435 <= 0.14181
0.0290429 <= 0.273763
```

Рисунок 26. Вывод неравенств для трёх случаев возмущенных входных данных

4. Для системы увеличенного размера (2n) тех же параметров генератора ошибок вычислить относительные погрешности входных данных, решения (матрица коэффициентов и вектор свободных членов заданы приближенно), □решения/ □данных, числа обусловленности.

```
n2 = 2*8;
A2 = zeros(n2, n2);
for i = 1:n2
    for j = 1:n2
        if (i == j)
            A2(i, j) = 1/i;
        else
            A2(i, j) = ((-1)^j)*j;
        end
    end
end
A2
```

Рисунок 27. Код создания и отображения матрицы A2

```
A2 =

   1.0000   2.0000  -3.0000   4.0000  -5.0000   6.0000  -7.0000   8.0000  -9.0000  10.0000 -11.0000  12.0000 -13.0000  14.0000 -15.0000  16.0000
  -1.0000   0.5000  -3.0000   4.0000  -5.0000   6.0000  -7.0000   8.0000  -9.0000  10.0000 -11.0000  12.0000 -13.0000  14.0000 -15.0000  16.0000
  -1.0000   2.0000   0.3333   4.0000  -5.0000   6.0000  -7.0000   8.0000  -9.0000  10.0000 -11.0000  12.0000 -13.0000  14.0000 -15.0000  16.0000
  -1.0000   2.0000  -3.0000   0.2500  -5.0000   6.0000  -7.0000   8.0000  -9.0000  10.0000 -11.0000  12.0000 -13.0000  14.0000 -15.0000  16.0000
  -1.0000   2.0000  -3.0000   4.0000   0.2000   6.0000  -7.0000   8.0000  -9.0000  10.0000 -11.0000  12.0000 -13.0000  14.0000 -15.0000  16.0000
  -1.0000   2.0000  -3.0000   4.0000  -5.0000   0.1667  -7.0000   8.0000  -9.0000  10.0000 -11.0000  12.0000 -13.0000  14.0000 -15.0000  16.0000
  -1.0000   2.0000  -3.0000   4.0000  -5.0000   6.0000   0.1429   8.0000  -9.0000  10.0000 -11.0000  12.0000 -13.0000  14.0000 -15.0000  16.0000
  -1.0000   2.0000  -3.0000   4.0000  -5.0000   6.0000  -7.0000   0.1250  -9.0000  10.0000 -11.0000  12.0000 -13.0000  14.0000 -15.0000  16.0000
  -1.0000   2.0000  -3.0000   4.0000  -5.0000   6.0000  -7.0000   8.0000   0.1111  10.0000 -11.0000  12.0000 -13.0000  14.0000 -15.0000  16.0000
  -1.0000   2.0000  -3.0000   4.0000  -5.0000   6.0000  -7.0000   8.0000  -9.0000   0.1000 -11.0000  12.0000 -13.0000  14.0000 -15.0000  16.0000
  -1.0000   2.0000  -3.0000   4.0000  -5.0000   6.0000  -7.0000   8.0000  -9.0000  10.0000   0.0909  12.0000 -13.0000  14.0000 -15.0000  16.0000
  -1.0000   2.0000  -3.0000   4.0000  -5.0000   6.0000  -7.0000   8.0000  -9.0000  10.0000 -11.0000   0.0833 -13.0000  14.0000 -15.0000  16.0000
  -1.0000   2.0000  -3.0000   4.0000  -5.0000   6.0000  -7.0000   8.0000  -9.0000  10.0000 -11.0000  12.0000   0.0769  14.0000 -15.0000  16.0000
  -1.0000   2.0000  -3.0000   4.0000  -5.0000   6.0000  -7.0000   8.0000  -9.0000  10.0000 -11.0000  12.0000 -13.0000   0.0714 -15.0000  16.0000
  -1.0000   2.0000  -3.0000   4.0000  -5.0000   6.0000  -7.0000   8.0000  -9.0000  10.0000 -11.0000  12.0000 -13.0000  14.0000   0.0667  16.0000
  -1.0000   2.0000  -3.0000   4.0000  -5.0000   6.0000  -7.0000   8.0000  -9.0000  10.0000 -11.0000  12.0000 -13.0000  14.0000 -15.0000   0.0625
```

Рисунок 28. Точная матрица A2

```
B2 = ones(n*2, 1);
B2(1, 1) = 2*8;
B2
```

Рисунок 29. Код создания и отображения вектора B2

```
B2 =

    16
     1
     1
     1
     1
     1
     1
     1
     1
     1
     1
     1
     1
     1
     1
     1
```

Рисунок 30. Точный вектор B2

```
74        X2 = linsolve(A2, B2)
```

Рисунок 31. Создание и отображения вектора X2

```
X2 =

    7.2125
    0.3834
   -0.1725
    0.1533
   -0.1106
    0.0986
   -0.0805
    0.0730
   -0.0631
    0.0581
   -0.0518
    0.0483
   -0.0440
    0.0413
   -0.0382
    0.0361
```

Рисунок 32. Точный вектор X2

```
noiseA2 = normrnd(0, 0.05/3, n2, n2);
noiseB2 = normrnd(0, 0.05/3, n2, 1);

An2 = A2 + noiseA2
Bn2 = B2 + noiseB2

Xn12 = linsolve(A2, Bn2)
Xn22 = linsolve(An2, B2)
Xn32 = linsolve(An2, Bn2)
```

Рисунок 33. Код создания и отображения возмущенных векторов X2, B2, матрицы A2

```
An2 =

    1.0242    2.0032   -3.0305    3.9822   -4.9964    5.9812   -6.9588    8.0009   -8.9887   10.0250  -10.9910   12.0078  -12.9799   13.9967  -15.0146   15.9912
   -1.0346    0.5165   -2.9928    4.0056   -4.9868    5.9907   -6.9916    7.9970   -9.0215   10.0023  -11.0137   11.9960  -13.0234   13.9938  -14.9892   15.9942
   -1.0029    2.0096    0.3304    4.0069   -4.9923    5.9865   -7.0137    8.0295   -8.9964    9.9736  -11.0179   12.0153  -12.9966   13.9859  -14.9825   16.0094
   -1.0109    2.0218   -2.9728    0.2526   -5.0102    6.0194   -6.9967    7.9582   -9.0261    9.9830  -11.0179   11.9849  -13.0332   14.0223  -14.9779   15.9866
   -0.9974    1.9878   -3.0227    4.0092    0.2374    6.0099   -7.0168    7.9924   -8.9869    9.9769  -10.9855   12.0024  -12.9995   13.9731  -15.0086   15.9691
   -1.0142    1.9856   -2.9838    3.9805   -4.9988    0.1707   -7.0102    8.0405   -9.0052   10.0159  -10.9836   12.0233  -12.9678   13.9918  -14.9996   16.0009
   -0.9953    1.9984   -2.9976    4.0168   -4.9856    6.0040    0.1319    7.9921   -8.9891    9.9900  -11.0293   11.9968  -12.9843   13.9838  -15.0027   15.9927
   -0.9900    2.0231   -2.9986    4.0019   -5.0069    5.9863   -7.0139    0.1157   -9.0090    9.9805  -11.0025   11.9859  -13.0001   13.9921  -15.0101   16.0047
   -0.9959    2.0217   -2.9881    4.0122   -5.0186    6.0166   -6.9937    7.9796    0.1166    9.9904  -10.9958   12.0324  -13.0056   13.9889  -14.9955   16.0014
   -1.0020    1.9979   -2.9801    3.9836   -4.9886    6.0058   -7.0186    8.0132   -8.9824    0.0861  -11.0102   11.9959  -12.9912   14.0080  -14.9940   16.0122
   -1.0215    1.9901   -3.0179    4.0009   -4.9827    5.9956   -6.9889    7.9648   -9.0330   10.0142    0.1036   12.0074  -12.9824   13.9826  -15.0059   16.0150
   -0.9990    1.9747   -2.9780    4.0146   -4.9696    5.9969   -6.9867    7.9867   -9.0311   10.0080  -10.9958    0.0699  -12.9998   13.9863  -14.9978   15.9779
   -1.0087    2.0112   -3.0202    4.0169   -5.0088    5.9983   -6.9827    8.0094   -9.0305   10.0050  -11.0025   12.0048    0.1038   14.0017  -14.9554   15.9746
   -1.0213    2.0004   -3.0179    3.9986   -5.0271    5.9852   -7.0003    8.0000   -8.9859   10.0069  -11.0093   12.0078  -13.0136    0.0895  -15.0001   16.0084
   -1.0009    1.9804   -3.0112    3.9691   -4.9730    6.0124   -6.9897    8.0104   -8.9932   10.0007  -10.9887   12.0001  -13.0333   13.9691    0.0785   16.0051
   -1.0035    1.9944   -2.9877    3.9817   -4.9956    6.0232   -6.9699    8.0021   -9.0117    9.9842  -10.9858   12.0054  -13.0125   14.0300  -15.0077    0.0468


Bn2 =

   16.0206
    1.0006
    1.0021
    1.0046
    1.0126
    1.0254
    0.9956
    0.9712
    1.0169
    0.9845
    1.0207
    0.9902
    1.0249
    0.9866
    0.9815
    1.0159
```

Рисунок 34. Возмущенные вектор B2 и матрица A2

```
Xn12 =

    7.2213
    0.3849
   -0.1728
    0.1529
   -0.1087
    0.0947
   -0.0815
    0.0771
   -0.0616
    0.0599
   -0.0502
    0.0493
   -0.0423
    0.0425
   -0.0396
    0.0353


Xn22 =

    7.1033
    0.2534
   -0.1821
    0.1461
   -0.1225
    0.0889
   -0.0917
    0.0888
   -0.0724
    0.0609
   -0.0422
    0.0517
   -0.0436
    0.0344
   -0.0402
    0.0372
```

```
Xn32 =

    7.1119
    0.2548
   -0.1824
    0.1455
   -0.1207
    0.0850
   -0.0928
    0.0929
   -0.0708
    0.0628
   -0.0405
    0.0528
   -0.0419
    0.0356
   -0.0417
    0.0363
```

Рисунок 35. Возмущенные векторы X2

```
err12 = norm((X2 - Xn12), 1)/norm(X2, 1)
err22 = norm((X2 - Xn22), 1)/norm(X2, 1)
err32 = norm((X2 - Xn32), 1)/norm(X2, 1)
```

Рисунок 36. Код вычисления и отображения относительных погрешностей векторов X2

```
err12 =

    0.0056


err22 =

    0.0279


err32 =

    0.0320
```

Рисунок 37. Относительные погрешности векторов X2

```
errInp12 = norm((B2 - Bn2), 1)/norm(B2, 1);
errInp22 = norm((A2 - An2), 1)/norm(A2, 1);
errInp32 = max(norm((A2 - An2), 1)/norm(A2, 1), norm((B2 - Bn2), 1)/norm(B2, 1));
dd12 = err12/errInp12
dd22 = err22/errInp22
dd32 = err32/errInp32
```

Рисунок 38. Код вычисления относительных погрешностей входных данных второй системы и решения/данных

```
dd12 =

    0.5876


dd22 =

   31.2722


dd32 =

    3.8150
```

Рисунок 39. решения/данных второй системы

```
condA2 = cond(A2, 1)
condAn2 = cond(An2, 1)

echoA2 = norm(inv(A2), 1)*norm(B2, 1)/norm(X2, 1)
echoAn12 = norm(inv(A2), 1)*norm(Bn2, 1)/norm(Xn12, 1)
echoAn22 = norm(inv(An2), 1)*norm(B2, 1)/norm(Xn22, 1)
echoAn32 = norm(inv(An2), 1)*norm(Bn2, 1)/norm(Xn32, 1)
```

Рисунок 40. Код вычисления и отображения числа обусловленности для приближенной матрицы A2 и ЕЧО для трёх случаев возмущения входных данных

```
condA2 =

  196.7015


condAn2 =

  201.4233


echoA2 =

    2.9313


echoAn12 =

    2.9236


echoAn22 =

    3.0376


echoAn32 =

    3.0295
```

Рисунок 41. Числа обусловленности для приближенной матрицы A2 и ЕЧО для трёх случаев возмущения входных данных

```
fprintf(1, '%g <= %g \n', err12, condAn2*errInp12)
fprintf(1, '%g <= %g \n', err22, condAn2*errInp22)
fprintf(1, '%g <= %g \n', err32, condAn2*errInp32)
```

Рисунок 42. Код для вывода неравенств для трёх случаев возмущенных входных данных второй системы

```
0.00379484 <= 1.22112
0.0542241 <= 0.251242
0.0533105 <= 1.22112
```

Рисунок 43. Вывод неравенств для трёх случаев возмущенных входных данных второй системы

5. Вычисленные значения представить в табличном виде. Проверить корректность вычисленных значений по соотношениям между ними. Сформулировать выводы по работе, в которых дать качественную оценку обусловленности задачи.

В таблицу записаны полученные данные для рассмотренных СЛАУ:

Название	**$A_{точное}$, $b_{прибл}$**	**$A_{прибл}$, $b_{точное}$**	**$A_{прибл}$, $b_{прибл}$**
	n = 8		
cond для $A_{точное}$	48.7831		
cond для $A_{прибл}$	49.7684		
Естественное число обусловленности	2.9851	3.0481	3.0742
$\delta_{вх}$	0.0053	0.0027	0.0053
$\delta_{реш}$	0.0032	0.0219	0.0193
Оценка относительной погрешности	0.259644	0.130283	0.259644
$\frac{\delta_{решения}}{\delta_{данных}}$	0.6012	8.2287	3.6347
	2n = 16		
cond для $A_{точное}$	196.7015		
cond для $A_{прибл}$	195.1720		
Естественное число обусловленности	2.9425	2.9909	3.0023
$\delta_{вх}$	0.0073	0.0011	0.0073
$\delta_{реш}$	0.0050	0.0268	0.0279
Оценка относительной погрешности	1.42629	0.205703	1.42629
$\frac{\delta_{решения}}{\delta_{данных}}$	0.6888	25.4762	3.8258

Вывод: анализ обусловленности задачи решения СЛАУ имеет важное значение в численных методах по нескольким причинам: на основе числа обусловленности матрицы системы можно понять, насколько чувствительно решение будет к погрешностям в данных, причем если число обусловленности лежит в пределах $1 \leq \text{cond} \leq 100$, то матрицу называют хорошо обусловленной, если cond ≥ 100 – плохо обусловленной; также анализ числа обусловленности позволяет дать оценку ожидаемой погрешности в решении СЛАУ на основе погрешности во входных данных. При увеличении размера матрицы изменяется число обусловленности – чем больше матрица, тем больше число обусловленности; также увеличивается размерность системы, и она становится более чувствительной к небольшим изменениям входных данных; к тому же для анализа обусловленности в больших системах требуются более ресурсоемкие вычисления. В рассматриваемом варианте задания число обусловленности составило 48.7831 для точной матрицы, поэтому, в целом, матрица хорошо обусловлена – небольшие изменения во входных данных не приводят к значительным изменениям в решении, о чем и свидетельствует $\delta_{реш}$.