In [2]:
```python
import numpy as np
```

Практическое задание по теме "Введение в линейную алгебру"

5.1.

Вектор – это частный случай матрицы 1xN и Nx1. Повторите материал для векторов, уделяя особое внимание умножению A∙B.
Вычислите, по возможности не используя программирование: (5E)-1 где E – единичная матрица размера 5x5.

$$5 \cdot \begin{vmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{vmatrix} - 1 = \begin{vmatrix} 4 & -1 & -1 & -1 & -1 \\ -1 & 4 & -1 & -1 & -1 \\ -1 & -1 & 4 & -1 & -1 \\ -1 & -1 & -1 & 4 & -1 \\ -1 & -1 & -1 & -1 & 4 \end{vmatrix}$$

In [3]:
```python
5*np.eye(5)-1
```

Out[3]:
```
array([[ 4., -1., -1., -1., -1.],
       [-1.,  4., -1., -1., -1.],
       [-1., -1.,  4., -1., -1.],
       [-1., -1., -1.,  4., -1.],
       [-1., -1., -1., -1.,  4.]])
```

5.2.

Вычислите определитель: $\begin{vmatrix} 1 & 2 & 3 \\ 4 & 0 & 6 \\ 7 & 8 & 9 \end{vmatrix} = A$

$$detA = 1 \cdot 0 \cdot 9 + 2 \cdot 6 \cdot 7 + 3 \cdot 4 \cdot 8 - 7 \cdot 0 \cdot 3 - 8 \cdot 6 \cdot 1 - 9 \cdot 4 \cdot 2 = 180 - 120 = \underline{60}$$

In [4]:
```python
A = [[1, 2, 3], [4, 0, 6], [7, 8, 9]]
np.linalg.det(A)
```

Out[4]:
```
59.999999999999986
```

5.3.

1. Вычислите матрицу, обратную данной: $\begin{vmatrix} 1 & 2 & 3 \\ 4 & 0 & 6 \\ 7 & 8 & 9 \end{vmatrix}$

$$A^{-1} = \frac{1}{detA} A^T_\bullet, detA \neq 0$$

Матрица алгебраических дополнений:

$$A_\bullet = \begin{vmatrix} -48 & 6 & 32 \\ 6 & -12 & 6 \\ 12 & 6 & -8 \end{vmatrix}$$

$$A^{-1} = \frac{1}{60} \cdot \begin{vmatrix} -48 & 6 & 12 \\ 6 & -12 & 6 \\ 32 & 6 & -8 \end{vmatrix} = \begin{vmatrix} -\frac{4}{5} & 0.1 & \frac{1}{5} \\ 0.1 & -\frac{1}{5} & 0.1 \\ \frac{8}{15} & 0.1 & -\frac{2}{15} \end{vmatrix}$$

In [5]:
```python
A_inv = np.linalg.inv(A)
A_inv
```

Out[5]:
```
array([[-0.8       ,  0.1       ,  0.2       ],
       [ 0.1       , -0.2       ,  0.1       ],
       [ 0.53333333,  0.1       , -0.13333333]])
```

In [6]:
```python
# A * A_inv
np.array([[round(k) for k in i] for i in (A @ A_inv)])
```

Out[6]:
```
array([[1, 0, 0],
       [0, 1, 0],
       [0, 0, 1]])
```

1. Приведите пример матрицы 4x4, ранг которой равен 1.

$$rank \begin{vmatrix} 1 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 \\ 7 & 0 & 0 & 0 \\ 10 & 0 & 0 & 0 \end{vmatrix} = 1$$

In [7]:
```python
np.linalg.matrix_rank([[1, 0, 0, 0],
                       [4, 0, 0, 0],
                       [7, 0, 0, 0],
                       [10, 0, 0, 0]])
```

Out[7]: 1

5.4.

Вычислите скалярное произведение двух векторов:

(1, 5) и (2, 8)

Ответ: 2 + 40 = 42

In [8]:
```
np.dot([1,5], [2,8])
```

Out[8]:
```
42
```

5.5.

Вычислите смешанное произведение трех векторов: (1, 5, 0), (2, 8, 7) и (7, 1.5, 3)

$$(\vec{a}, \vec{b}, \vec{c}) = \begin{vmatrix} a_x & a_y & a_z \\ b_x & b_y & b_z \\ c_x & c_y & c_z \end{vmatrix} = a_x b_y c_z + a_y b_z c_x + a_z b_x c_y - a_z b_y c_x - c_y b_z b_x - c_z b_x a_y$$

$$\begin{vmatrix} 1 & 5 & 0 \\ 2 & 8 & 7 \\ 7 & 1.5 & 3 \end{vmatrix} = 1 \cdot 8 \cdot 3 + 5 \cdot 7 \cdot 7 + 0 \cdot 2 \cdot 1.5 - 0 \cdot 8 \cdot 7 - 5 \cdot 2 \cdot 3 - 1 \cdot 7 \cdot 1.5 = \underline{228.5}$$

In [9]:
```
f = [[1, 5, 0], [2, 8, 7], [7, 1.5, 3]]
np.linalg.det(f)
```

Out[9]:
```
228.50000000000009
```

In [10]:
```
a = np.array([1, 5, 0], float)
b = np.array([2, 8, 7], float)
c = np.array([7, 1.5, 3], float)
v = np.cross(a, b)
print (np.inner(v, c))
```

```
228.5
```

Практическое задание по теме "Системы линейных алгебраических уравнений"

1.

Решите линейную систему: $\begin{vmatrix} 1 & 2 & 3 \\ 4 & 0 & 6 \\ 7 & 8 & 9 \end{vmatrix} \cdot X = \begin{vmatrix} 12 \\ 2 \\ 1 \end{vmatrix}, \qquad X = \begin{vmatrix} x \\ y \\ z \end{vmatrix}$

$$X = A^{-1} \cdot \begin{vmatrix} 12 \\ 2 \\ 1 \end{vmatrix} = \begin{vmatrix} 12 \cdot -\frac{4}{5} + 2 \cdot 0.1 + 1 \cdot \frac{1}{5} \\ 12 \cdot 0.1 + 2 \cdot -\frac{1}{5} + 1 \cdot 0.1 \\ 12 \cdot \frac{8}{15} + 2 \cdot 0.1 - 1 \cdot \frac{2}{15} \end{vmatrix} = \begin{vmatrix} -9.2 \\ 0.9 \\ 6.4(6) \end{vmatrix}$$

In [11]:
```
A_inv @ [[12], [2], [1]]
```

Out[11]:
```
array([[-9.2       ],
       [ 0.9       ],
       [ 6.46666667]])
```

2.

Найдите псевдорешение:

$x + 2y - z = 1$

$3x - 4y = 7$

$8x - 5y + 2z = 12$

$2x - 5z = 7$

$11x + 4y - 7z = 15$

In [12]:
```
A = np.array([[1, 2, -1],
              [3, -4, 0],
              [8, -5, 2],
              [2, 0, -5],
              [11, 4, -7]])
B = np.array([[1],
             [7],
             [12],
             [7],
             [15]])
np.linalg.lstsq(A, B, rcond=None)
```

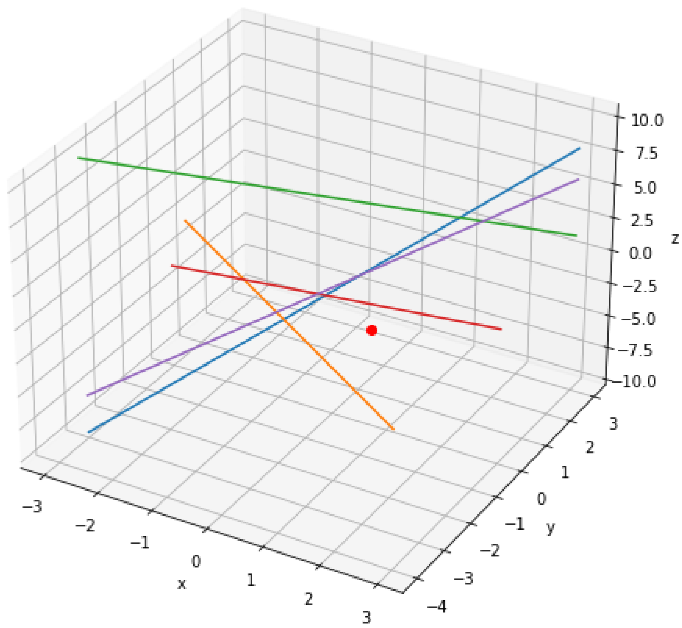
Out[12]:
```
(array([[ 1.13919353],
        [-0.90498444],
        [-0.9009803 ]]),
 array([0.71523211]),
```

```
 3,
 array([15.2817306 ,  9.59852942,  3.65197794]))
```

In [13]:
```python
np.dot(A, [1.13919353, -0.90498444, -0.9009803])
```

Out[13]:
```
array([ 0.23020495,  7.03751835, 11.83650984,  6.78328856, 15.21805317])
```

In [14]:
```python
import matplotlib.pyplot as plt
%matplotlib inline
x = np.linspace(-3, 3, 201)
y = np.linspace(-3, 3, 201)
z = 2*y+x-1

fig = plt.figure(figsize=(8, 8))

ax = fig.gca(projection='3d')

ax.plot(x, y, z)
ax.plot(x, -((7+3*y)/4), [0]*201)
ax.plot(x, y, 6+2.5*y-4*x)
ax.plot(x,[0]*201, (2*x-7)/5)
ax.plot(x, y, (11*x + 4*y - 5)/7)
ax.plot(1.13919353, -0.90498444, -0.9009803, 'ro')

ax.set_xlabel('x')
ax.set_ylabel('y')
ax.set_zlabel('z')
plt.show()
```



Ответ: $X = \begin{vmatrix} 1.14 \\ -0.9 \\ -0.9 \end{vmatrix}$

3.

Сколько решений имеет линейная система: $\begin{vmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{vmatrix} \cdot X = \begin{vmatrix} 12 \\ 2 \\ 1 \end{vmatrix}$

Если ноль - то измените вектор правой части так, чтобы система стала совместной, и решите её.

$$A = \begin{vmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{vmatrix} \qquad B = \begin{vmatrix} 12 \\ 2 \\ 1 \end{vmatrix} \qquad (A|B) = \begin{vmatrix} 1 & 2 & 3 & 12 \\ 4 & 5 & 6 & 2 \\ 7 & 8 & 9 & 1 \end{vmatrix}$$

Чтобы система была совместной, ранги основной и расширенной матриц должны совпадать.
Ранг матрицы можно вычислить с помощью метода Гаусса:

$$(A) = \begin{vmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{vmatrix} \sim \begin{vmatrix} 1 & 2 & 3 \\ 0 & -3 & -6 \\ 0 & -6 & -12 \end{vmatrix} \sim \begin{vmatrix} 1 & 2 & 3 \\ 0 & -3 & -6 \\ 0 & 0 & 0 \end{vmatrix} \Rightarrow rank(A) = 2$$

$$(A|B) = \begin{vmatrix} 1 & 2 & 3 & 12 \\ 4 & 5 & 6 & 2 \\ 7 & 8 & 9 & 1 \end{vmatrix} \sim \begin{vmatrix} 1 & 2 & 3 & 12 \\ 0 & -3 & -6 & -46 \\ 0 & -6 & -12 & -83 \end{vmatrix} \sim \begin{vmatrix} 1 & 2 & 3 & 12 \\ 0 & -3 & -6 & -46 \\ 0 & 0 & 0 & 9 \end{vmatrix} \Rightarrow rank(A|B) = 3$$

$rank(A) \neq rank(A|B) \Rightarrow$ система несовместна.

$$B = \begin{vmatrix} 1 \\ 1 \\ 1 \end{vmatrix}$$

$$(A|B) = \begin{vmatrix} 1 & 2 & 3 & 1 \\ 4 & 5 & 6 & 1 \\ 7 & 8 & 9 & 1 \end{vmatrix} \sim \begin{vmatrix} 1 & 2 & 3 & 1 \\ 0 & -3 & -6 & -3 \\ 0 & -6 & -12 & -6 \end{vmatrix} \sim \begin{vmatrix} 1 & 2 & 3 & 1 \\ 0 & -3 & -6 & -3 \\ 0 & 0 & 0 & 0 \end{vmatrix} \Rightarrow rank(A|B) = 2$$

$rank(A) = rank(A|B)$, система совместна. Ранги матриц меньше количества столбцов A $\Rightarrow$ система имеет бесконечное мно

Общее решение:

$x + 2y + 3z = 1$

$-3y - 6z = -3$

$y = 1 - 2z \qquad x = 1 - 2(1 - 2z) - 3z = z - 1$

$$X = \begin{vmatrix} z - 1 \\ 1 - 2z \\ z \end{vmatrix}$$

4.

Вычислите LU-разложение матрицы: $\begin{vmatrix} 1 & 2 & 3 \\ 2 & 16 & 21 \\ 4 & 28 & 73 \end{vmatrix}$

In [15]:

```
import scipy
import scipy.linalg

A = np.array([ [1, 2, 3], [2, 16, 21], [4, 28, 73]])
P, L, U = scipy.linalg.lu(A)

print("P = ", P)
print("L = ", L)
print("U = ", U)
```

```
P =  [[0. 1. 0.]
 [0. 0. 1.]
 [1. 0. 0.]]
L =  [[ 1.    0.    0.  ]
 [ 0.25  1.    0.  ]
 [ 0.5  -0.4   1.  ]]
U =  [[  4.    28.    73.  ]
 [  0.    -5.   -15.25]
 [  0.     0.   -21.6 ]]
```

5.

Найдите нормальное псевдорешение недоопределенной системы:

$x + 2y - z = 1$

$8x - 5y + 2z = 12$

In [16]:

```
A = [[1, 2, -1], [8, -5, 2]]
B = [[1], [12]]
C = [[1, 2, -1, 1], [8, -5, 2, 12]]
```

In [17]:

```
if np.linalg.matrix_rank(C) == np.linalg.matrix_rank(A):
    print('Система совместна')
else:
    print("Система несовместна")
```

```
Система совместна
```

In [18]:

```
A @ np.linalg.lstsq(A, B, rcond=None)[0]
```

Out[18]:

```
array([[ 1.],
       [12.]])
```

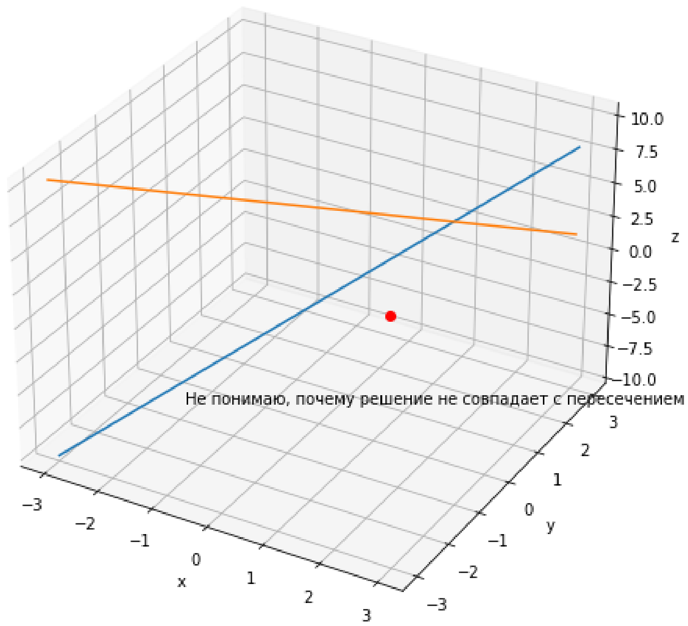
Невязка равна нулю => псевдорешение с минимальной длиной невязки называется нормальным *псевдорешением* => $x_0 = \begin{bmatrix} 1 \\ 12 \end{bmatrix}$

является нормальным псевдорешением

In [19]:

```
import matplotlib.pyplot as plt
%matplotlib inline
x = np.linspace(-3, 3, 201)
y = np.linspace(-3, 3, 201)
z = 2*y+x-1

fig = plt.figure(figsize=(8, 8))

ax = fig.gca(projection='3d')

ax.plot(x, y, z)
ax.plot(x, y, 6+2.5*y-4*x)
ax.plot([ 1.38191882],
```

```
         [-0.18081181],
         [ 0.0202952 ], 'ro')
# ax.plot(1.13919353, -0.90498444, -0.9009803, 'ro')

ax.set_xlabel('x')
ax.set_ylabel('y')
ax.set_zlabel('z')
ax.text(0, -4, 0, 'Не понимаю, почему решение не совпадает с пересечением')
plt.show()
```



6.

Найдите одно из псевдорешений вырожденной системы: $\begin{vmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{vmatrix} \cdot X = \begin{vmatrix} 2 \\ 5 \\ 11 \end{vmatrix}$

Попробуйте также отыскать и нормальное псевдорешение.

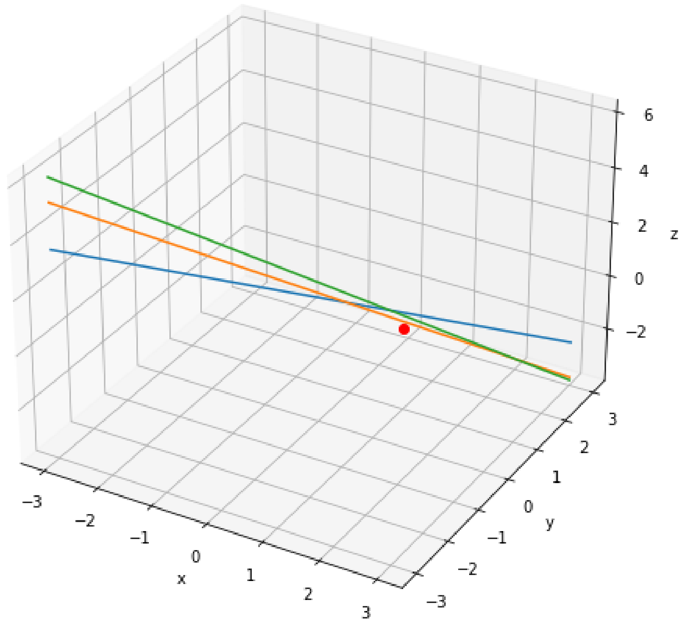
In [20]:
```
A = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])
B = np.array([[2], [5], [11]])
C = np.concatenate((A, B), axis=1)
np.linalg.matrix_rank(A), np.linalg.matrix_rank(C)
```

Out[20]:
```
(2, 3)
```

In [25]:
```
np.linalg.lstsq(A, B, rcond=None)
```

Out[25]:
```
(array([[ 1.25],
        [ 0.5 ],
        [-0.25]]),
 array([], dtype=float64),
 2,
 array([1.68481034e+01, 1.06836951e+00, 1.47280825e-16]))
```

In [28]:
```
# здесь я чувствую себя неуверенно
%matplotlib inline
x = np.linspace(-3, 3, 201)
y = np.linspace(-3, 3, 201)
z = (2-2*y-x)/3

fig = plt.figure(figsize=(8, 8))

ax = fig.gca(projection='3d')

ax.plot(x, y, z)
ax.plot(x, y, (5-5*y-4*x)/6)
ax.plot(x, y, (11-8*y-7*x)/9)
ax.plot([ 1.25],
        [ 0.5 ],
        [-0.25], 'ro')

ax.set_xlabel('x')
ax.set_ylabel('y')
ax.set_zlabel('z')
plt.show()
```


x
y
z
−3
−2
−1
0
1
2
3
−3
−2
−1
0
1
2
3
−2
0
2
4
6