In [1]:
```python
import numpy as np
import scipy.linalg as sp
from sympy import *
init_printing(use_unicode=True)
```

Задача 1

Линейный оператор $\varphi \in Hom(\mathbb{R}^4, \mathbb{R}^2)$ задан своей матрицей A_φ в стандратном базисе. Найти $\varphi^{-1}(x)$, если

$$A_\varphi = \begin{pmatrix} -1 & 3 & 9 & 10 \\ 1 & -2 & -6 & -7 \end{pmatrix}, \quad x = \begin{pmatrix} 16 \\ -11 \end{pmatrix}$$

На первой строке введите вектора, которые войдут в ответ без свободных коэффициентов, на второй, которые войдут со свободными коэффициентами, в случае если таковых векторов нет, вводите []

Для ответа $\varphi^{-1}(x) = \{ \begin{pmatrix} 1 \\ 2.023 \\ 3 \end{pmatrix} + c_1 \begin{pmatrix} 3.04 \\ 2.023 \\ 3 \end{pmatrix} + c_2 \begin{pmatrix} 5.719 \\ 2 \\ 1 \end{pmatrix} \mid c_1, c_2 \in \mathbb{R}\}$

Пример ввода: $[1, 2.02, 3]$
$[3.04, 2.02, 3; 5.72, 2, 1]$

Для ответа $\varphi^{-1}(x) = \{ \begin{pmatrix} 1 \\ -1.347 \\ 2.111 \end{pmatrix} \}$

Пример ввода: $[1, -1.35, 2.11]$
$[]$

Сохранить

In [5]:
```python
A = Matrix([[-1, 3, 9, 10], [1, -2, -6, -7]])
A_conj = Matrix([[-1, 3, 9, 10, 16], [1, -2, -6, -7, -11]])
A.nullspace(), A.rref()
```

Out[5]:

$$\left(\left[\begin{bmatrix} 0 \\ -3 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ -3 \\ 0 \\ 1 \end{bmatrix} \right], \left(\begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 3 & 3 \end{bmatrix}, (0, 1) \right) \right)$$

In [4]:
```python
A_conj.rref()
```

Out[4]:

$$\left(\begin{bmatrix} 1 & 0 & 0 & -1 & -1 \\ 0 & 1 & 3 & 3 & 5 \end{bmatrix}, (0, 1) \right)$$

Задача 2

Оператор $\varphi \in Hom(\mathbb{R}^3, \mathbb{R}^3)$ задан своей матрицей A_φ в паре базисов $\{e\}_{i=1}^3$ и $\{h\}_{i=1}^3$, являющихся базисами пространств области определения области значения оператора соответственно.

Найти матрицу этого оператора $\tilde{A}_\varphi$ в паре базисов$\{\tilde{e}\}_{i=1}^3$ и $\{\tilde{h}\}_{i=1}^3$, если

$$A_\varphi = \begin{pmatrix} 1 & -2 & 0 \\ -1 & 2 & -1 \\ 2 & -5 & 3 \end{pmatrix}$$

$$e_0 = \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix},\ e_1 = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix},\ e_2 = \begin{pmatrix} 1 \\ -1 \\ 1 \end{pmatrix}$$

$$h_0 = \begin{pmatrix} 1 \\ -1 \\ 1 \end{pmatrix},\ h_1 = \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix},\ h_2 = \begin{pmatrix} -2 \\ 3 \\ -3 \end{pmatrix}$$

$$\tilde{e}_0 = \begin{pmatrix} 1 \\ -2 \\ 3 \end{pmatrix},\ \tilde{e}_1 = \begin{pmatrix} -2 \\ 5 \\ -8 \end{pmatrix},\ \tilde{e}_2 = \begin{pmatrix} -2 \\ 2 \\ -1 \end{pmatrix}$$

$$\tilde{h}_0 = \begin{pmatrix} 1 \\ -2 \\ 0 \end{pmatrix},\ \tilde{h}_1 = \begin{pmatrix} -2 \\ 5 \\ -1 \end{pmatrix},\ \tilde{h}_2 = \begin{pmatrix} -2 \\ 3 \\ 2 \end{pmatrix}$$

Для ответа $\tilde{A}_\varphi = \begin{pmatrix} 1 & 2.034 & -1.436 \\ 7.348 & 2 & 1 \\ 3.055 & 1.155 & 3 \end{pmatrix}$

In [6]:

```
A = Matrix([[1, -2, 0], [-1, 2, -1], [2, -5, 3]])
# B = T_y^-1 dot A dot T_x
# T^-1 = ~E^-1 dot E

# T_y section
yE_tilda = Matrix([[1, -2, -2], [-2, 5, 3], [0, -1, 2]])
yE = Matrix([[1, 1, -2], [-1, 0, 3], [1, -1, -3]])
T_y_inv = yE_tilda**(-1) * yE

# T_x section
# T_x = E^-1 * ~E
xE = Matrix([[1, 1, 1], [1, 2, -1], [2, 3, 1]])
xE_tilda = Matrix([[1, -2, -2], [-2, 5, 2], [3, -8, -1]])
T_x = xE**(-1) * xE_tilda

T_y_inv * A * T_x
```

Out[6]:

$$\begin{bmatrix} 508 & -1593 & 375 \\ 133 & -417 & 98 \\ 94 & -295 & 70 \end{bmatrix}$$

Задача 3

Найти спектральное разложение оператора $\varphi \in Hom(\mathbb{R}^3, \mathbb{R}^3)$, заданного матрицей в стандартном базисе.

$$A_\varphi = \begin{pmatrix} 2 & 0 & 0 \\ 1 & 1 & 1 \\ 2 & -2 & 4 \end{pmatrix}$$

На отдельных строках введите собственные числа и матрицу оператора проекции на соответствующие ему собственные подпространства. В разложении каждому собственному числу должна соответствовать ровна одна матрица оператора проектирования, которая проецирует на всё собственное подпространство. Для ответа

$$A_\varphi = 3 \begin{pmatrix} 3 & 4 \\ -1.234 & 1.211 \end{pmatrix} - 6 \begin{pmatrix} 8.43 & 4.21 \\ -2.239 & 1.23 \end{pmatrix}$$

Пример ввода: $3\ [3, 4; -1.23, 1.21]$
$-\ 6\ [8.43, 4.21; -2.24, 1.23]$

Сохранить

```
In [47]: A = Matrix([[2, 0, 0], [1, 1, 1], [2, -2, 4]])
T = Matrix([[1, -1, 0], [1, 0, 1/2], [0, 1, 1]])
S = T**(-1)
eig_two_first = T * Matrix([[2, 0, 0], [0, 0, 0], [0, 0, 0]]) * S
eig_two_second = T * Matrix([[0, 0, 0], [0, 2, 0], [0, 0, 0]]) * S
eig_3 = T * Matrix([[0, 0, 0], [0, 0, 0], [0, 0, 3]]) * S
eig_two_first / 2, eig_two_second / 2, eig_3 / 3
```

Out[47]:

$$\left(\begin{bmatrix} -1.0 & 2.0 & -1.0 \\ -1.0 & 2.0 & -1.0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 2.0 & -2.0 & 1.0 \\ 0 & 0 & 0 \\ -2.0 & 2.0 & -1.0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 \\ 1.0 & -1.0 & 1.0 \\ 2.0 & -2.0 & 2.0 \end{bmatrix} \right)$$

Задача 4

Найдите матрицу оператора $f(\varphi) = 3(\sin(\varphi))^2$, если оператор $\varphi \in Hom(\mathbb{R}^4, \mathbb{R}^4)$ задан своей матрицей в стандартном базисе

$$A_\varphi = \begin{pmatrix} 2 & 1 & 0 & 0 \\ 5 & 3 & 2 & 1 \\ -7 & -4 & -1 & -1 \\ -7 & -4 & -2 & 0 \end{pmatrix}$$

В качестве ответа ввести матрицу искомого оператора
Для ответа

$$A_{f(\varphi)} = \begin{pmatrix} 0.2 & 1.041 & 1.5 \\ 1 & 2 & 3 \\ 1.5 & 2 & 3.136 \end{pmatrix}$$

введите
Пример ввода: [0.2, 1.04, 1.5; 1, 2, 3; 1.5, 2, 3.14]

Сохранить

In [52]:

```python
def to_string(a):
        result = ""
        for i in range(len(a)):
                for j in range(len(a[i])):
                        if j != len(a) - 1:
                                result += str(a[i][j]) + ", "
                        else:
                                result += str(a[i][j]) + "; "
        return "[" + result[:-2] + "]"
```

In [77]:

```python
A = np.array([[2, 1, 0, 0], [5, 3, 2, 1], [-7, -4, -1, -1], [-7, -4, -2, 0]])
Asin = sp.sinm(A)
res = 3 * (Asin @ Asin)
to_string(np.round(res, 6))
```

Out[77]:

```
'[-2.638531, -1.017429, -2.496881, -1.248441; 21.130104, 11.325326, 7.952666, 3.976333;
-11.604603, -7.166248, -0.834683, -1.479452; -11.604603, -7.166248, -2.958904, 0.64476
8]'
```

In [88]:

```python
t = np.array([[6, 1, 1, -2/3], [-6, 5, 0, 2/3], [-6, -7, 0, 1], [-6, -7, 0, 0]])
j = np.array([[1, 1, 0, 0], [0, 1, 1, 0], [0, 0, 1, 0], [0, 0, 0, 1]])
jnew = sp.sinm(j)
jlast = 3*(jnew @ jnew)
t1 = np.array([[0, -7/72, 7/108, -29/216], [0, 1/12, -1/18, -1/36], [1, 1/2, 1/3, 1/6],
np.round(t @ jlast @ t1, 6)
```

Out[88]:

```
array([[ -2.638531,  -1.017429,  -2.496881,  -1.248441],
       [ 21.130104,  11.325326,   7.952666,   3.976333],
       [-11.604603,  -7.166248,  -0.834683,  -1.479452],
       [-11.604603,  -7.166248,  -2.958904,   0.644768]])
```