```python
import sympy
from sympy import Eq, S, latex, plot_implicit, Matrix, Symbol, symbols, simplify, expand, collect, solve, solveset, zeros, ones
from sympy import pi as Pi
from sympy import cos as Cos
from sympy import sin as Sin
from sympy.plotting.plot import plot3d
import matplotlib.pyplot as plt
%matplotlib inline
```

Занятие 17

Алгебра

Поверхности второго порядка

Уравнение поверхности второго порядка: $$ a_{11}x^2+a_{22}y^2+a_{33}z^2+2a_{12}xy+2a_{13}xz+2a_{23}yz+2a_{14}x+2a_{24}y+2a_{34}z+a_{44} $$ В матричном виде: $$ X^TAX+2bX+c,\quad X=\left(\begin{matrix}x\\y\\z\end{matrix}\right),\quad A=\left(\begin{matrix}a_{11}&a_{12}&a_{13}\\a_{12}&a_{22}&a_{23}\\a_{13}&a_{23}&a_{33}\end{matrix}\right),\quad b=\left(\begin{matrix}a_{14}&a_{24}&a_{34}\end{matrix}\right),\quad c=a_{44} $$ $$ X^TA_1X,\quad X=\left(\begin{matrix}x\\y\\z\\1\end{matrix}\right), A=\left(\begin{matrix}a_{11}&a_{12}&a_{13}&a_{14}\\a_{12}&a_{22}&a_{23}&a_{24}\\a_{13}&a_{23}&a_{33}&a_{34}\\a_{14}&a_{23}&a_{33}&a $$

Пример 1.

Записать уравнение поверхности второго порядка $x^2-2y^2-z^2+2xy+4xz+6x-8y+4z-12=0$ в матричном виде двумя способами. Вначале построим матрицу A квадратичной формы, входящей в уравнение, и вектор b: $$ A=\left(\begin{matrix}1&1&2\\1&-2&0\\2&0&-1\end{matrix}\right),\quad b=\left(\begin{matrix}3&-4&2\end{matrix}\right) $$ Проверим, что эта матрица задает нашу поверхность второго порядка, для этого вычислим $$ X^TAX+2bX+c,\quad X=\left(\begin{matrix}x\\y\\z\end{matrix}\right), \quad c=-12 $$

In [2]:
```python
x, y, z = symbols('x y z')
A = Matrix(((1, 1, 2), (1, -2, 0), (2, 0, -1)))
b = Matrix((3, -4, 2)).T
c = -12
X = Matrix((x, y, z))
u = X.T*A*X
v = b*X
display(A, u, v)
eq1 = simplify(expand(u[0] + 2*v[0] + c))
display(Eq(eq1, 0))
```

$\displaystyle \left[\begin{matrix}1 & 1 & 2\\1 & -2 & 0\\2 & 0 & -1\end{matrix}\right]$

$\displaystyle \left[\begin{matrix}x \left(x + y + 2 z\right) + y \left(x - 2 y\right) + z \left(2 x - z\right)\end{matrix}\right]$

$\displaystyle \left[\begin{matrix}3 x - 4 y + 2 z\end{matrix}\right]$

$\displaystyle x^{2} + 2 x y + 4 x z + 6 x - 2 y^{2} - 8 y - z^{2} + 4 z - 12 = 0$

Теперь составим расширенную матрицу A_1 и проверим, что получилось то же самое уравнение:

In [3]:
```python
A1 = A.row_join(b.T).col_join(b.row_join(Matrix((c,)))) # (c,) - tuple, состоящий из одного элемента
display(A1)
X1 = X.col_join(Matrix((1,)))
display(Eq(simplify(expand(X1.T*A1*X1))[0], 0))
```

$\displaystyle \left[\begin{matrix}1 & 1 & 2 & 3\\1 & -2 & 0 & -4\\2 & 0 & -1 & 2\\3 & -4 & 2 & -12\end{matrix}\right]$

$\displaystyle x^{2} + 2 x y + 4 x z + 6 x - 2 y^{2} - 8 y - z^{2} + 4 z - 12 = 0$

Классификация поверхностей второго порядка

Для классификации поверхностей второго порядка нужно привести их уравнение к каноническому виду, переходя к базису из собственных векторов или выделяя полные квадраты.

15 типов поверхностей второго порядка:

1. Эллипсоид $$ \frac{x^2}{a^2}+\frac{y^2}{b^2}+\frac{z^2}{c^2}=1 $$
2. Однополостной гиперболоид $$ \frac{x^2}{a^2}+\frac{y^2}{b^2}-\frac{z^2}{c^2}=1 $$
3. Двуполостной гиперболоид $$ \frac{x^2}{a^2}+\frac{y^2}{b^2}-\frac{z^2}{c^2}=-1 $$
4. Конус $$ \frac{x^2}{a^2}+\frac{y^2}{b^2}-\frac{z^2}{c^2}=0 $$
5. Эллиптический параболоид $$ \frac{x^2}{a^2}+\frac{y^2}{b^2}=2z $$
6. Гиперболический параболоид $$ \frac{x^2}{a^2}-\frac{y^2}{b^2}=2z $$
7. Эллиптический цилиндр $$ \frac{x^2}{a^2}+\frac{y^2}{b^2}=1 $$
8. Гиперболический цилиндр $$ \frac{x^2}{a^2}-\frac{y^2}{b^2}=1 $$
9. Параболический цилиндр $$ y^2=2px $$
10. Пара пересекающихся плоскостей $$ y^2-k^2x^2=0,quad k\neq0 $$
11. Пара параллельных плоскостей $$ y^2-k^2=0,quad k\neq0 $$
12. Плоскость $$ y^2=0 $$
13. Прямая $$ x^2+y^2=0 $$
14. Одна точка $$ x^2+y^2+z^2=0 $$
15. Пустое множество $$ x^2+y^2+z^2=-1,\quad x^2+y^2=-1,\quad x^2=-1 $$

Пример 2.

Определить тип поверхности второго порядка, заданной уравнением $$ 14x^2 − 6\sqrt2 xy − 6\sqrt2xz + 13y^2 + 2yz + 13z^2 - 4 = 0 $$ Составим матрицу 3 порядка, соответствующую нашему уравнению, заметим, что вектор b в нашем случае нулевой. Найдем собственные векторы матрицы и перейдем к базису из собственных векторов.

In [4]:
```python
A = Matrix(((14, -3*2**S.Half, -3*2**S.Half), (-3*2**S.Half, 13, 1), (-3*2**S.Half, 1, 13)))
X = Matrix((x, y, z))
display(A, Eq(simplify(expand(X.T*A*X))[0] - 4, 0))
P = Matrix([])
for item in A.eigenvects():
    degree = item[1]
    for i in range(degree):
        P = P.row_join(item[2][i].normalized())
A2 = P.T*A*P
x1, y1, z1 = symbols('x1 y1 z1')
X_new = Matrix((x1, y1, z1))
q_f2 = simplify(expand(X_new.T*A2*X_new))[0]
display(A2, q_f2)
```

$\displaystyle \left[\begin{matrix}14 & - 3 \sqrt{2} & - 3 \sqrt{2}\\- 3 \sqrt{2} & 13 & 1\\- 3 \sqrt{2} & 1 & 13\end{matrix}\right]$

$\displaystyle 14 x^{2} - 6 \sqrt{2} x y - 6 \sqrt{2} x z + 13 y^{2} + 2 y z + 13 z^{2} - 4 = 0$

$\displaystyle \left[\begin{matrix}8 & 0 & 0\\0 & 12 & 0\\0 & 0 & 20\end{matrix}\right]$

$\displaystyle 8 x_{1}^{2} + 12 y_{1}^{2} + 20 z_{1}^{2}$

Свободный член уравнения не изменился в результате такого преобразования квадратичной формы, поэтому оно запишется в виде

In [5]:
```python
Eq(q_f2, 4)
```

Out[5]: $\displaystyle 8 x_{1}^{2} + 12 y_{1}^{2} + 20 z_{1}^{2} = 4$

Поделим обе части уравнения на 4 и получим каноническое уравнение эллипсоида.

In [6]:
```python
eq2 = Eq(q_f2/4, 1)
display(eq2)
```

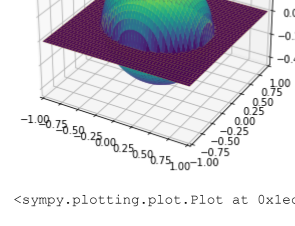
$\displaystyle 2 x_{1}^{2} + 3 y_{1}^{2} + 5 z_{1}^{2} = 1$

Получился эллипсоид. Выразим z и изобразим на графике

In [7]:
```python
Z = solve(eq2, z1)
display(*Z)
plot3d(*Z, (x1, -1, 1), (y1, -1, 1))
```

$\displaystyle - \frac{\sqrt{- 10 x_{1}^{2} - 15 y_{1}^{2} + 5}}{5}$

$\displaystyle \frac{\sqrt{- 10 x_{1}^{2} - 15 y_{1}^{2} + 5}}{5}$



Out[7]: `<sympy.plotting.plot.Plot at 0x1ecef00f940>`

Пример 3.

Определить тип поверхности второго порядка, заданной уравнением $$ 3x^2 - 7y^2 + 3z^2 + 8xy - 8yz - 8xz + 10x - 14y - 6z - 8 = 0 $$ Вначале приведем к каноническому виду квадратичную форму $3x^2 - 7y^2 + 3z^2 + 8xy - 8yz - 8xz$.

In [8]:
```python
A3 = Matrix(((3, 4, -4), (4, -7, -4), (-4, -4, 3)))
X3 = Matrix((x, y, z))
display(A3, simplify(expand(X3.T*A3*X3))[0])
P = Matrix([])
for item in A3.eigenvects():
    degree = item[1]
    for i in range(degree):
        P = P.row_join(item[2][i].normalized())
A3_new = P.T*A3*P
x1, y1, z1 = symbols('x1 y1 z1')
X3_new = Matrix((x1, y1, z1))
q_f_new = simplify(expand(X3_new.T*A3_new*X3_new))[0]
display(A3_new, q_f_new)
```

$\displaystyle \left[\begin{matrix}3 & 4 & -4\\4 & -7 & -4\\-4 & -4 & 3\end{matrix}\right]$

$\displaystyle 3 x^{2} + 8 x y - 8 x z - 7 y^{2} - 8 y z + 3 z^{2}$

$\displaystyle \left[\begin{matrix}-9 & 0 & 0\\0 & -1 & 0\\0 & 0 & 9\end{matrix}\right]$

$\displaystyle - 9 x_{1}^{2} - y_{1}^{2} + 9 z_{1}^{2}$

Введем вектор-столбец линейной формы $b = (5, -7, -3)$ и умножим его слева на транспонированную матрицу перехода к базису из собственных векторов.

In [9]:
```python
b = Matrix((5, -7, -3))
b_new = P.T*b
display(b_new)
```

$\displaystyle \left[\begin{matrix}- 6 \sqrt{2}\\\sqrt{2}\\-3\end{matrix}\right]$

Составляем новое уравнение, не содержащее попарных произведений переменных, т.е. x_1y_1, x_1z_1 и y_1z_1:

In [10]:
```python
u_new = q_f_new + 2*(b_new.T*X3_new)[0]
display(u_new)
```

$\displaystyle - 9 x_{1}^{2} - 12 \sqrt{2} x_{1} - y_{1}^{2} + 2 \sqrt{2} y_{1} + 9 z_{1}^{2} - 6 z_{1}$

Теперь нужно выделить полные квадраты.

Выделим полный квадрат в выражении $ax^2 + bx$: $$ ax^2 + 2bx = a\left(x^2 + 2 \frac{b}{a}x + \frac{b^2}{a^2} - \frac{b^2}{a^2}\right)= a\left(x + \frac{b}{a}\right)^2 - \frac{b^2}{a}. $$ В результате таких замен получается квадратичная форма с теми же коэффициентами, что и до выделения квадратов, но становится нулевой линейная часть, а свободный член вычисляется по остаткам $- \frac{b^2}{a}$.

Заметим, что вектор b_{new} содержит коэффициенты при линейных членах квадратичной формы, а диагональные элементы матрицы A_{new} являются коэффициентами при соответствующих квадратах.

In [11]:
```python
c_new = -8
for i in range(3):
    c_new -= b_new[i]**2/(A3_new[i, i])
display(c_new)
```

$\displaystyle 1$

Новые переменные x_2, y_2, z_2 связаны с x_1, y_1, z_1 соотношениями $$ x_2 = x_1 + \frac{b_1}{a_{11}}, \quad y_2 = y_1 + \frac{b_2}{a_{22}}, \quad z_2 = z_1 + \frac{b_3}{a_{33}}, $$ где (b_1, b_2, b_2) - вектор b_{new}, a_{ii}, $i = 1, 2, 3$ - диагональные элементы матрицы A_{new}.

В новых переменных x_2, y_2, z_2 уравнение поверхности такое:

In [12]:
```python
x2, y2, z2 = symbols('x2 y2 z2')
X3_last = Matrix((x2, y2, z2))
Eq((X3_last.T*A3_new*X3_last)[0] + c_new, 0)
```

Out[12]: $\displaystyle - 9 x_{2}^{2} - y_{2}^{2} + 9 z_{2}^{2} + 1 = 0$

Для получения канонического уравнения осталось перенести свободный член в правую часть и поделить обе части на свободный член с противоположным знаком, получим

In [13]:
```python
eq3 = Eq((X3_last.T*A3_new*X3_last)[0]/(-c_new), 1)
display(eq3)
```

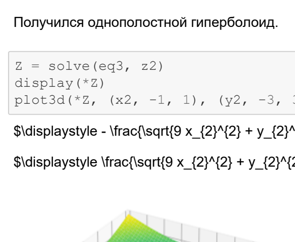
$\displaystyle 9 x_{2}^{2} + y_{2}^{2} - 9 z_{2}^{2} = 1$

Получился однополостной гиперболоид.

In [14]:
```python
Z = solve(eq3, z2)
display(*Z)
plot3d(*Z, (x2, -1, 1), (y2, -3, 3))
```

$\displaystyle - \frac{\sqrt{9 x_{2}^{2} + y_{2}^{2} - 1}}{3}$

$\displaystyle \frac{\sqrt{9 x_{2}^{2} + y_{2}^{2} - 1}}{3}$



Out[14]: `<sympy.plotting.plot.Plot at 0x1ecf0b92208>`

Пример 4.

Привести уравнение поверхности второго порядка $x^2 + y^2 + 4z^2 + 2xy + 4xz + 4yz - 6z - 5 = 0$ к каноническому виду. Построить график.

In [15]:
```python
A4 = Matrix(((1, 1, 2), (1, 1, 2), (2, 2, 4)))
X4 = Matrix((x, y, z))
P = Matrix([])
for item in A4.eigenvects():
    degree = item[1]
    for i in range(degree):
        P = P.row_join(item[2][i].normalized())
A4_new = P.T*A4*P
x1, y1, z1 = symbols('x1 y1 z1')
X4_new = Matrix((x1, y1, z1))
q_f_new = simplify(expand(X4_new.T*A4_new*X4_new))[0]
display(A4_new, q_f_new)
```

$\displaystyle \left[\begin{matrix}0 & 0 & 0\\0 & 0 & 0\\0 & 0 & 6\end{matrix}\right]$

$\displaystyle 6 z_{1}^{2}$

У матрицы квадратичной формы одно ненулевое собственное значение. Пересчитаем вектор коэффициентов линейной формы:

In [16]:
```python
b = Matrix((0, 0, -3))
b_new = P.T*b
display(b_new)
u_new = q_f_new + 2*(b_new.T*X4_new)[0]
display(u_new)
```

$\displaystyle \left[\begin{matrix}0\\- \frac{3 \sqrt{5}}{5}\\- \sqrt{6}\end{matrix}\right]$

$\displaystyle - \frac{6 \sqrt{5} y_{1}}{5} + 6 z_{1}^{2} - 2 \sqrt{6} z_{1}$

Видим, что только переменная z_1 входит в уравнение и во второй степени, и в первой, выделим полный квадрат и проведем замену переменной. При этом коэффициенты квадратичной формы останутся теми же, изменится только свободный член.

In [17]:
```python
c_new = -5
c_new -= b_new[2]**2/(A4_new[2, 2])
print('c_new =', c_new)
b_last = b_new
b_last[2] = 0
x2, y2, z2 = symbols('x2 y2 z2')
X4_last = Matrix((x2, y2, z2))
eq4 = Eq((X4_last.T*A4_new*X4_last)[0] + 2*(b_last.T*X4_last)[0] + c_new, 0)
display(eq4)
```

```
c_new = -6
```

$\displaystyle - \frac{6 \sqrt{5} y_{2}}{5} + 6 z_{2}^{2} - 6 = 0$

Получился параболический цилиндр. Чтобы поучить каноническое уравнение, достаточно выразить z_2^2 и провести замену переменной y_2, избавляющую от свободного члена:

In [18]:
```python
Z = solve(eq4, z2**2)
display(*Z)
```

$\displaystyle \frac{\sqrt{5} y_{2}}{5} + 1$

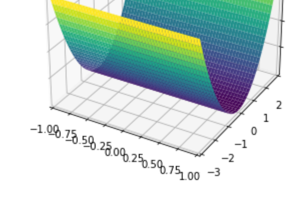
Для того, чтобы избавиться от свободного члена, проведем замену переменной $y_2 = y_3 - \sqrt{5}$:

In [19]:
```python
y3 = Symbol('y3')
eq4 = Eq(z2**2, simplify(expand(Z[0].subs(y2, y3 - 5**S.Half))))
eq4
```

Out[19]: $\displaystyle z_{2}^{2} = \frac{\sqrt{5} y_{3}}{5}$

Выразим с помощью solve y_3 из полученного уравнения и построим график:

In [20]:
```python
Y = solve(eq4, y3)
plot3d(*Y, (x2, -1, 1), (z2, -3, 3))
```



Out[20]: `<sympy.plotting.plot.Plot at 0x1ecf0de57b8>`

Пример 5.

Привести уравнение поверхности второго порядка $x^2 + 5y^2 + z^2 + 2xy + 6xz + 2yz - 2x + 6y + 2z + 1 = 0$ к каноническому виду. Построить график.

In [21]:
```python
A4 = Matrix(((1, 1, 3), (1, 5, 1), (3, 1, 1)))
X4 = Matrix((x, y, z))
b = Matrix((-1, 3, 1))
u = simplify(expand(X4.T*A4*X4))[0] + 2*(b.T*X4)[0]
display(u)
P = Matrix([])
for item in A4.eigenvects():
    degree = item[1]
    for i in range(degree):
        P = P.row_join(item[2][i].normalized())
A4_new = P.T*A4*P
x1, y1, z1 = symbols('x1 y1 z1')
X4_new = Matrix((x1, y1, z1))
q_f_new = simplify(expand(X4_new.T*A4_new*X4_new))[0]
display(A4_new, q_f_new)
```

$\displaystyle x^{2} + 2 x y + 6 x z - 2 x + 5 y^{2} + 2 y z + 6 y + z^{2} + 2 z$

$\displaystyle \left[\begin{matrix}-2 & 0 & 0\\0 & 3 & 0\\0 & 0 & 6\end{matrix}\right]$

$\displaystyle - 2 x_{1}^{2} + 3 y_{1}^{2} + 6 z_{1}^{2}$

In [22]:
```python
b_new = P.T*b
display(b_new)
u_new = q_f_new + 2*(b_new.T*X4_new)[0]
display(u_new)
```

$\displaystyle \left[\begin{matrix}\sqrt{2}\\- \sqrt{3}\\\sqrt{6}\end{matrix}\right]$

$\displaystyle - 2 x_{1}^{2} + 2 \sqrt{2} x_{1} + 3 y_{1}^{2} - 2 \sqrt{3} y_{1} + 6 z_{1}^{2} + 2 \sqrt{6} z_{1}$

In [23]:
```python
c_new = 1
for i in range(3):
    c_new -= b_new[i]**2/(A4_new[i, i])
display(c_new)
```

$\displaystyle 0$

In [24]:
```python
b_last = zeros(3, 1)
x2, y2, z2 = symbols('x2 y2 z2')
X4_last = Matrix((x2, y2, z2))
eq4 = Eq((X4_last.T*A4_new*X4_last)[0] + 2*(b_last.T*X4_last)[0] + c_new, 0)
display(eq4)
```

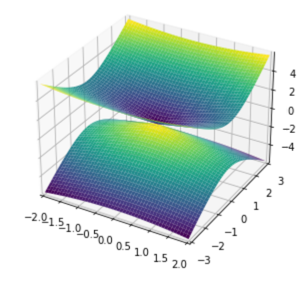
$\displaystyle - 2 x_{2}^{2} + 3 y_{2}^{2} + 6 z_{2}^{2} = 0$

Получился конус. Чтобы поучить каноническое уравнение, достаточно выразить x_2^2:

In [25]:
```python
X = solve(eq4, x2)
display(*X)
plot3d(*X, (y2, -2, 2), (z2, -3, 3))
```

$\displaystyle - \frac{\sqrt{6 y_{2}^{2} + 12 z_{2}^{2}}}{2}$

$\displaystyle \frac{\sqrt{6 y_{2}^{2} + 12 z_{2}^{2}}}{2}$



Out[25]: `<sympy.plotting.plot.Plot at 0x1ecf10bbda0>`