

## 0 Задания к уроку 2

### 1. Задание

Даны два вектора в трехмерном пространстве: (10,10,10) и (0,0,-10)

1) Найдите их сумму. (на листочке)

$$B1 + B2 = (10+0, 10+0, 10-10) = (10,10,0)$$

```
a = np.array([10,10,10])
b = np.array([0,0,-10])
a + b

array([10, 10,  0])
```

2) Напишите код на Python, реализующий расчет длины вектора, заданного его координатами. (в программе)

```
def vector_length(x1, x2):
    v_l = math.sqrt((x1**2 + x2**2))
    print(v_l)
```

```
vector_length(-3, 4)
```

5.0

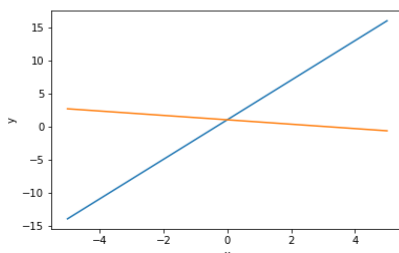
### 2. Задание (на листочке)

Почему прямые не кажутся перпендикулярными? (см.ролик)

```
In [1]: %matplotlib inline
import numpy as np
import matplotlib.pyplot as plt
import math
```

```
In [2]: x = np.linspace(-5, 5, 21)
y = 3*x+1
y2 = (-1/3)*x+1
plt.plot(x,y)
plt.plot(x,y2)
plt.xlabel("x")
plt.ylabel("y")
```

Out[2]: <matplotlib.text.Text at 0x6aa80f0>

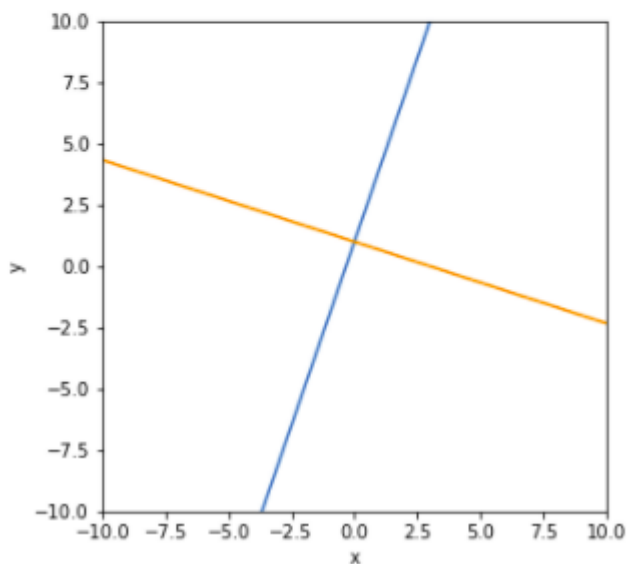


Из-за того, что график сильно растянут по оси X. Если сделать координатную сетку более однородной, то визуальное представление изменится:

```
plt.figure(figsize = (5, 5))
plt.ylim(-10, 10)
plt.xlim(-10, 10)
```

```
x = np.linspace(-10, 10, 21)
y = 3*x+1
y2 = (-1/3)*x+1
plt.plot(x,y)
plt.plot(x,y2)
plt.xlabel("x")
plt.ylabel("y")
```

Text(0, 0.5, 'y')



### 3. Задание (в программе)

Напишите код на Python, реализующий построение графиков:

1. окружности,

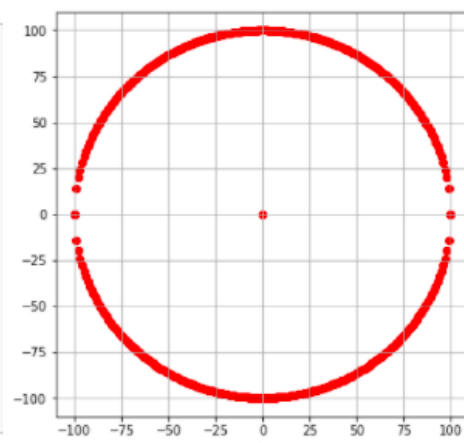
эта окружность рисуется точками, которые находятся из уравнения окружности

```
r = 100

plt.figure(figsize = (6, 6))
plt.xlabel('x')
plt.ylabel('y')
plt.grid(True)

plt.scatter(0, 0, color='r')

for x in range(-100,101):
    y = np.sqrt(r ** 2 - x ** 2)
    plt.scatter(x, y, color='r')
    y = -np.sqrt(r ** 2 - x ** 2)
    plt.scatter(x, y, color='r')
```



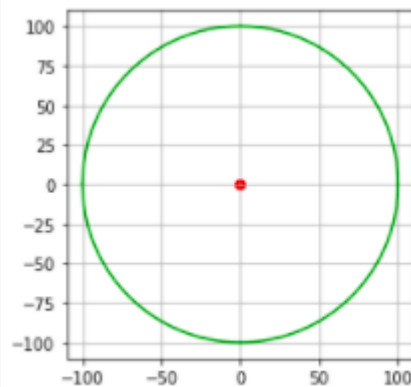
А этот график я подсмотрели и немножко поменяла:

```
x = []
x2 = []
y = []
y2 = []

for i in range(1000):
    r = 100
    x.append(-i)
    x2.append(i)
    y.append(np.sqrt(r ** 2 - i ** 2))
    y2.append(-np.sqrt(r ** 2 - i ** 2))

plt.scatter(0, 0, color='r')
plt.grid(True)

plt.plot(x, y, color='g')
plt.plot(x, y2, color='g')
plt.plot(x2, y2, color='g')
plt.plot(x2, y, color='g')
plt.axis('scaled')
plt.show()
```



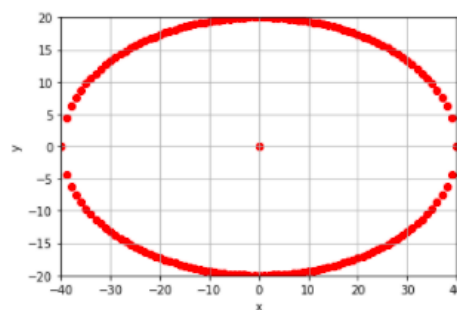
## 2. эллипса,

```
a = 40
b = 20

plt.ylim(-b, b)
plt.xlim(-a, a)
plt.xlabel('x')
plt.ylabel('y')
plt.grid(True)

plt.scatter(0, 0, color='r')

for x in range(-100,101):
    y=np.sqrt(b ** 2 - (b ** 2 * (x ** 2 / a ** 2)))
    plt.scatter(x, y, color='r')
    y=-np.sqrt(b ** 2 - (b ** 2 * (x ** 2 / a ** 2)))
    plt.scatter(x, y, color='r')
```



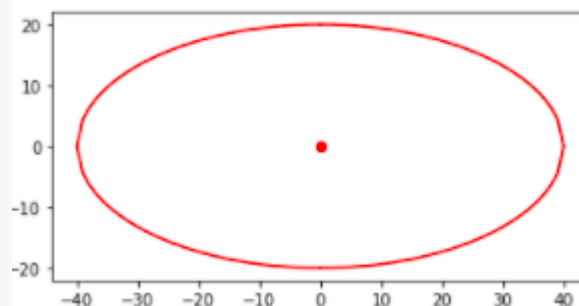
И второй вариант:

```
x = []
x2 = []
y = []
y2 = []

plt.scatter(0, 0, color='r')

for i in range(1000):
    a = 40
    b = 20
    x.append(-i)
    x2.append(i)
    y.append(np.sqrt(b ** 2 - (b ** 2 * (i ** 2 / a ** 2))))
    y2.append(-np.sqrt(b ** 2 - (b ** 2 * (i ** 2 / a ** 2))))

plt.plot(x, y, color='r')
plt.plot(x, y2, color='r')
plt.plot(x2, y2, color='r')
plt.plot(x2, y, color='r')
plt.axis('scaled')
plt.show()
```

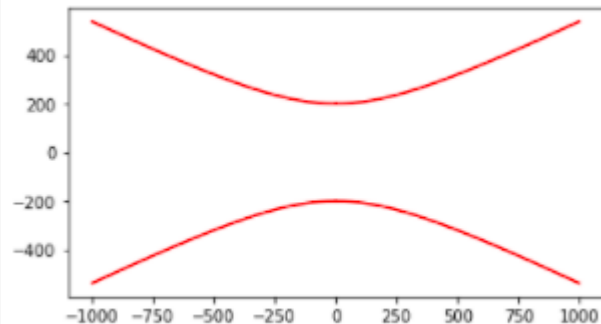


## 3. гиперболы.

```
x = []
x2 = []
y = []
y2 = []

for i in range(1000):
    a = 400
    b = 200
    x.append(-i)
    x2.append(i)
    y.append(np.sqrt(b ** 2 + (b ** 2 * (i ** 2 / a ** 2))))
    y2.append(-np.sqrt(b ** 2 + (b ** 2 * (i ** 2 / a ** 2))))

plt.plot(x, y, color='r')
plt.plot(x, y2, color='r')
plt.plot(x2, y2, color='r')
plt.plot(x2, y, color='r')
plt.axis('scaled')
plt.show()
```



#### 4. Задание (на листочке)

1) Пусть задана плоскость:

$$A \cdot x + B \cdot y + C \cdot z + D = 0$$

Напишите уравнение плоскости, параллельной данной и проходящей через начало координат.

Плоскость проходит через начало координат в точке (0,0,0). Подставим эти координаты в уравнение  $A \cdot 0 + B \cdot 0 + C \cdot 0 + D = 0$

Из полученного уравнения очевидно, что  $D=0$ .

$$A \cdot x + B \cdot y + C \cdot z = 0$$

2) Пусть задана плоскость:  $A_1x + B_1y + C_1z + D_1 = 0$   
и прямая:

$$\frac{x - x_1}{x_2 - x_1} = \frac{y - y_1}{y_2 - y_1} = \frac{z - z_1}{z_2 - z_1}$$

Как узнать, принадлежит прямая плоскости или нет?

Поставить координаты (x, y, z) и (x<sub>1</sub>, y<sub>1</sub>, z<sub>1</sub>) из уравнения прямой в уравнение плоскости. Если уравнению плоскости получится тождество, значит прямая принадлежит плоскости.

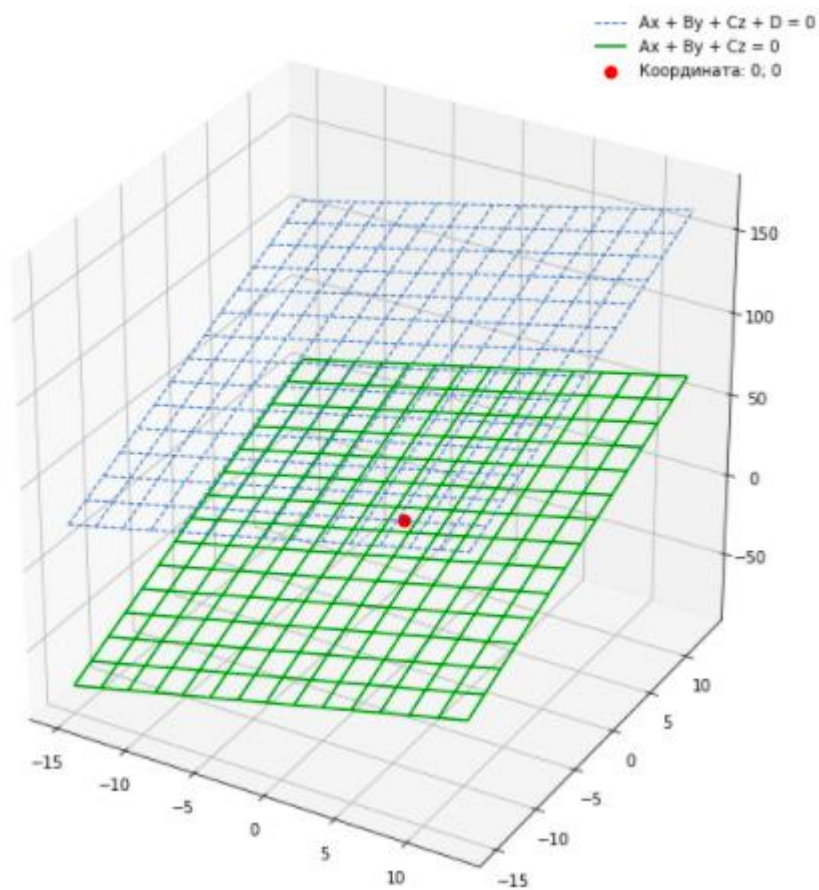
#### 4. Задание (в программе)

1) Нарисуйте трехмерный график двух параллельных плоскостей.

```
from pylab import *
from mpl_toolkits.mplot3d import Axes3D
fig = figure(figsize = (8, 8))
ax = Axes3D(fig)
X = np.arange(-15, 15, 2)
Y = np.arange(-15, 15, 2)

X, Y = np.meshgrid(X, Y)

Z = 2*X + 3*Y + 100
Z2 = 2*X + 3*Y
ax.plot_wireframe(X, Y, Z, label='Ax + By + Cz + D = 0', linestyle='--', linewidth=1)
ax.plot_wireframe(X, Y, Z2, colors='green', label='Ax + By + Cz = 0')
ax.scatter(0, 0, 0, 'z', 50, 'red', label='Координата: 0; 0; 0')
plt.legend(frameon=False)
show()
```



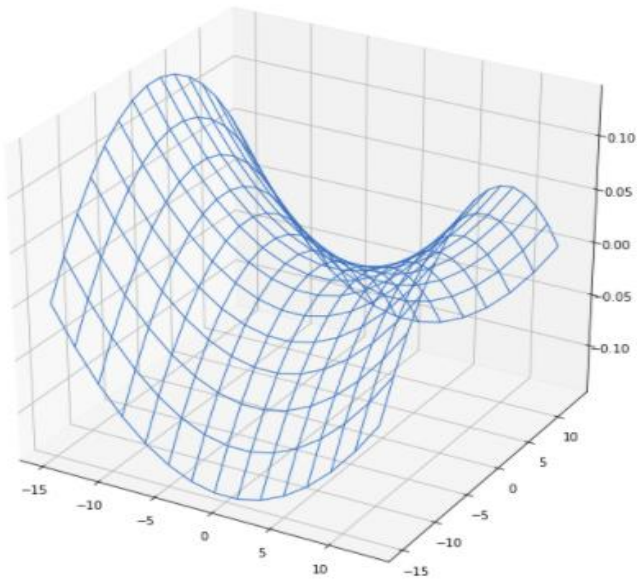
- 2) Нарисуйте трехмерный график двух любых поверхностей второго порядка.

```
fig = figure(figsize = (8, 8))
ax = Axes3D(fig)

a = 40
b = 40
x, y = np.meshgrid((np.arange(-15, 15, 2)), (np.arange(-15, 15, 2)))
z = x**2/a**2 - y**2/b**2

ax.plot_wireframe(x, y, z, linewidth=1)

show()
```



```
fig = figure(figsize = (8, 8))
ax = Axes3D(fig)

a = 40
b = 40
x, y = np.meshgrid((np.arange(-15, 15, 2)), (np.arange(-15, 15, 2)))
z = x**2/a**2 + y**2/b**2

ax.plot_wireframe(x, y, z, linewidth=1, color='g')

show()
```

