

Функции в Python

Функция - это именованный блок кода, к которому можно обратиться из любого места программы. У функции есть имя и список входных параметров, а также возвращаемое значение.

Функция позволяет использовать в программе один и тот же фрагмент кода несколько раз.

Объявление функции в Python выглядит так:

```
def function_name(argument1, argument2, ...):
    # код функции
# def - DEclare Function - "объявить функцию"
# function_name - имя функции
# (argument1, argument2, ...) - список аргументов, поступающих
на вход функции при ее вызове
# тело функции - это весь код, который идет после двоеточия

# Объявление функции

def hello(name):
    print('Hello, ' + name)

# Вызовы функции

hello('Max')
hello('Ivan')
hello('Alex')
hello('Kate')

# Вывод

>> Hello, Max
>> Hello, Ivan
>>
>>
```

Оператор `return` возвращает значение из функции.

Представьте себе обычный калькулятор. Вы вводите первое число, операцию и второе число. Калькулятор возвращает нам результат операции над этими числами. Всё просто, не так ли? Функции точно так же умеют возвращать значение с помощью специального оператора `return`.

```
# Объявление функции

def sum2(a, b):
    return a + b

# Вызовы функции
```

```

s1 = sum2(10, 2)
s2 = sum2(108, 100)
s3 = sum2(3, 1)

print(f's1 = {s1}')
print(f's2 = {s2}')
print(f's3 = {s3}')
```

>> s1 = 12
>> s2 = 208
>> s3 = 4

```

# Функция умножения двух чисел

def mult2(a, b):
    return a * b

# Вызовем нашу функцию

m1 = mult2(10, 2)
m2 = mult2(108, 100)
m3 = mult2(3, 1)

print(f'm1 = {m1}')
```

>> = 20
>> = 10800
>> =3

Для параметров функции можно указывать значения по умолчанию. Это дает возможность вызывать функцию с меньшим числом параметров.

```

# Аргумент name по умолчанию равен 'world'

def hello(name='world'):
    print('Hello, ' + name)

hello()
hello('Ivan')
```

>> Hello, world
>>

Примеры

1. Квадрат

```

# Написать функцию square(), вычисляющую квадрат числа.

def square number
```

```
    return number * number # Возвращаем результат работы функции
    обратно в программу

a = square(2)
print(a)

>> 4
```

2. Периметр

```
# Напишите функцию perimetr, вычисляющую периметр прямоугольника
со сторонами a и b.

def perimetr(a, b):
    return 2 * (a + b)

p = perimetr(4, 3)
print(p)

>> 14
```

3. Четное число

```
# Напишите функцию isEven, возвращающую True, если число четное,
и False, если - нечетное.

def isEven(x):
    return x % 2 == 0

print(isEven(10))
print(isEven(11))

>> True
>>
```

4. Сумма списка

```
# Напишите функцию amountList, которая возвращает сумму всех
элементов списка.

def amountList(lst):
    amount = 0
    for x in lst:
        amount += x
    return amount

print(amountList([1, 2, 3]))

mylist = [1, 2, 4, 8, 16]
s = amountList(mylist)

print(f'Сумма списка {mylist} равна {s}')
```

```
>> 6
>> Сумма списка [1, 2, 4, 8, 16] равна 31
```

5. Фибоначчи

```
# Напишите функцию fib, которая возвращает n-ное число
# Фибоначчи.
# Последовательность Фибоначчи выглядит так: 1 1 2 3 5 8 13 21
# 34

def fib(n):
    a, b = 0, 1
    if n == 0: return 0
    for i in range(1, n):
        a, b = b, a + b
    return b

print(fib(2))
print(fib(3))
print(fib(4))
print(fib(5))
print(fib(10))

>> 1
>> 2
>> 3
>> 5
>> 55
```

6. Факториал

```
# Напишите функцию fact, вычисляющую значение факториала числа
# N.
# Факториал числа - это произведение всех чисел от 1 до N.
# Например, факториал числа 5 равен 120 (5! = 120).

def fact(n):
    result = 1
    while n > 1:
        result *= n
        n -= 1
    return result

print(fact(2))
print(fact(3))
print(fact(4))
print(fact(5))

>> 2
>> 6
>> 24
>> 120
```

Визуализация данных с matplotlib

Библиотека `matplotlib` содержит большой набор инструментов для двумерной графики. Она проста в использовании и позволяет получать графики высокого качества. В этом разделе мы рассмотрим наиболее распространенные типы диаграмм и различные настройки их отображения.

Модуль `matplotlib.pyplot` предоставляет процедурный интерфейс к (объектно-ориентированной) библиотеке `matplotlib`, который во многом копирует инструменты пакета [MATLAB](#). Инструменты модуля `pyplot` де-факто являются стандартным способом работы с библиотекой `matplotlib`, поэтому мы органичимся рассмотрением этого пакета.

Двумерные графики

`pyplot.plot`

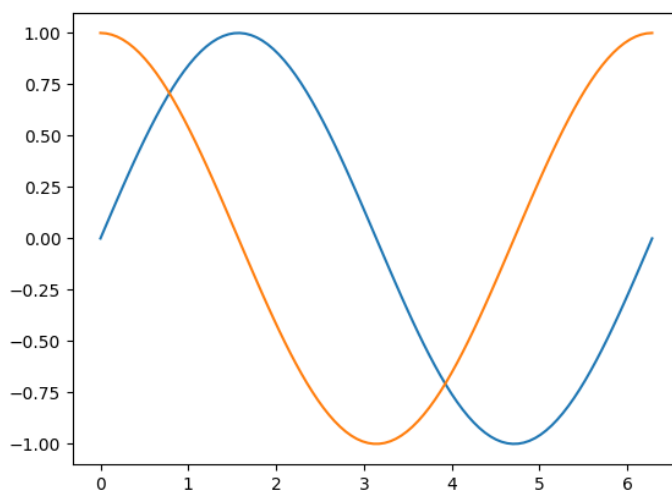
Нарисовать графики функций `sin` и `cos` с `matplotlib.pyplot` можно следующим образом:

```
import numpy as np
import matplotlib.pyplot as plt

phi = np.linspace(0, 2.*np.pi, 100)
plt.plot(phi, np.sin(phi))
plt.plot(phi, np.cos(phi))

plt.show()
```

В результате получаем



Мы использовали функцию `plot`, которой передали два параметра — списки значений по горизонтальной и вертикальной осям. При последовательных вызовах функции `plot` графики строятся в одних осях, при этом происходит автоматическое переключение цвета.

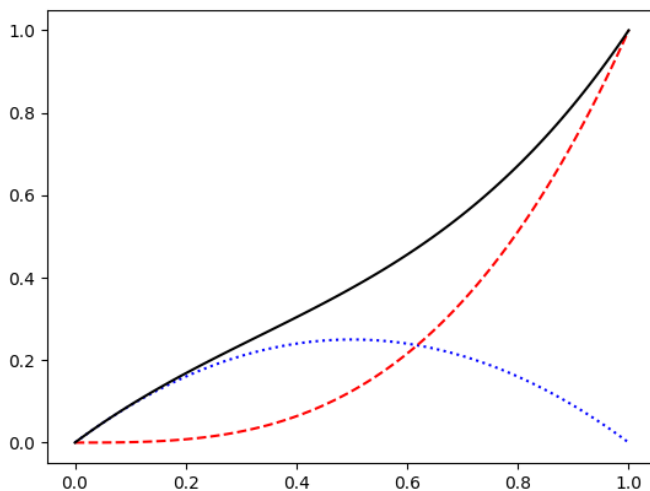
Строковый параметр

```
fmt = '[marker][line][color]'
```

функции `plot` позволяет задавать тип маркера, тип линии и цвет. Приведем несколько примеров:

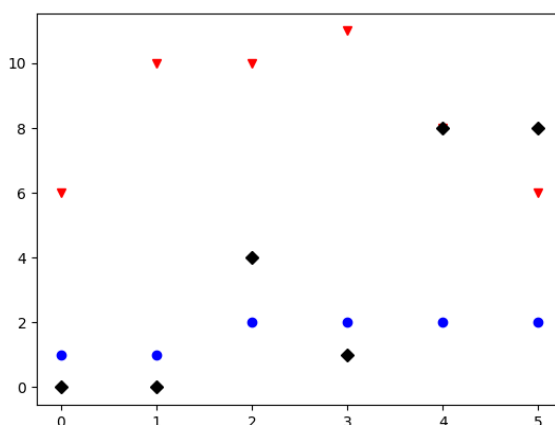
```
x = np.linspace(0, 1, 100)
f1 = 0.25 - (x - 0.5)**2
f2 = x**3
plt.plot(x, f1, ':b')      # пунктирная синяя линия
plt.plot(x, f2, '--r')    # штрихованная красная линия
plt.plot(x, f1+f2, 'k')   # черная непрерывная линия

plt.show()
```



```
rg = np.random.Generator(np.random.PCG64())
plt.plot(rg.binomial(10, 0.3, 6), 'ob')  # синие круги
plt.plot(rg.poisson(7, 6), 'vr')        # красные треугольники
plt.plot(rg.integers(0, 10, 6), 'Dk')    # черные ромбы

plt.show()
```



Из последнего примера видно, что если в функцию `plot` передать только один список `y`, то он будет использован для значений по вертикальной оси. В качестве значений по горизонтальной оси будет использован `range(len(y))`.

Более тонкую настройку параметров можно выполнить, передавая различные именованные аргументы, например:

- `marker: str` — тип маркера
- `markersize: float` — размер маркера
- `linestyle: str` — тип линии
- `linewidth: float` — толщина линии
- `color: str` — цвет

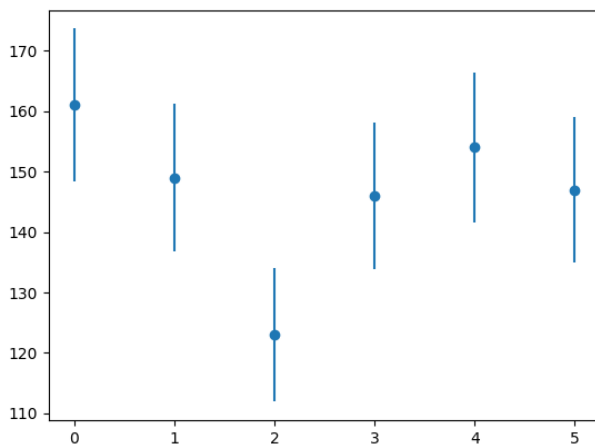
Полный список доступных параметров можно найти [в документации](#).

pyplot.errorbar

Результаты измерений в физике чаще всего представлены в виде величин с ошибками.

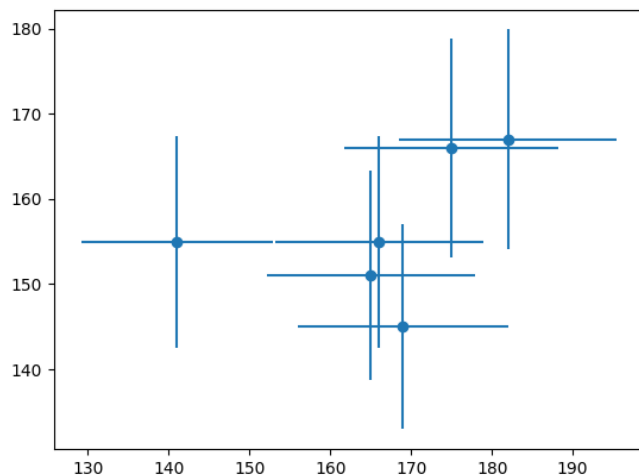
Функция `plt.errorbar` позволяет отображать такие данные:

```
rg = np.random.Generator(np.random.PCG64(5))
x = np.arange(6)
y = rg.poisson(149, x.size)
plt.errorbar(x, y, yerr=np.sqrt(y), marker='o',
             linestyle='none')
plt.show()
```



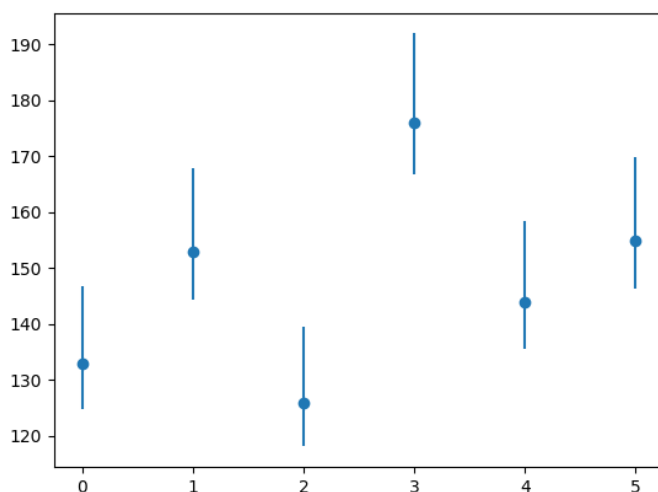
Ошибки можно задавать и для значений по горизонтальной оси:

```
rg = np.random.Generator(np.random.PCG64(5))
N = 6
x = rg.poisson(169, N)
y = rg.poisson(149, N)
plt.errorbar(x, y, xerr=np.sqrt(x), yerr=np.sqrt(y), marker='o',
             linestyle='none')
plt.show()
```



Ошибки измерений могут быть асимметричными. Для их отображения в качестве параметра `yerr` (или `xerr`) необходимо передать кортеж из двух списков:

```
rg = np.random.Generator(np.random.PCG64(11))
N = 6
x = np.arange(N)
y = rg.poisson(149, N)
yerr = [
    0.7*np.sqrt(y),
    1.2*np.sqrt(y)
]
plt.errorbar(x, y, yerr=yerr, marker='o', linestyle='none')
plt.show()
```



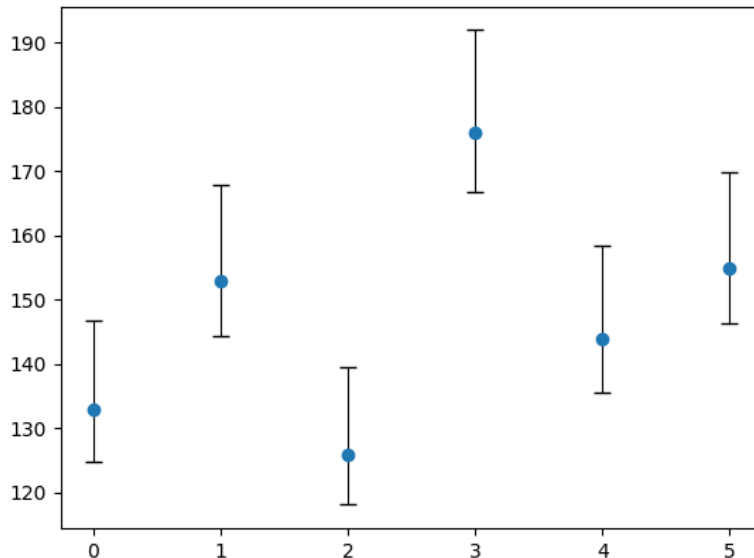
Функция `pyplot.errorbar` поддерживает настройку отображения графика с помощью параметра `fmt` и всех именованных параметров, которые доступны в функции `pyplot`. Кроме того, здесь появляются параметры для настройки отображения линий ошибок (“усов”):

- `ecolor: str` — цвет линий ошибок

- `elinewidth: float` — ширина линий ошибок
- `capsize: float` — длина “колпачков” на концах линий ошибок
- `capthick: float` — толщина “колпачков” на концах линий ошибок

и некоторые другие. Изменим параметры отрисовки данных из предыдущего примера:

```
# ...
plt.errorbar(x, y, yerr=yerr, marker='o', linestyle='none',
             ecolor='k', elinewidth=0.8, capsize=4, capthick=1)
plt.show()
```



Настройки отображения

Наши графики все еще выглядят довольно наивно. В этой части мы рассмотрим различные настройки, которые позволят достичь качества оформления диаграмм, соответствующего, например, публикациям в рецензируемых журналах.

Диапазон значений осей

Задавать диапазон значений осей в `matplotlib` можно несколькими способами. Например, так:

```
pyplot.xlim([0, 200]) # диапазон горизонтальной оси от 0 до 200
pyplot.ylim([0, 1])   # диапазон вертикальной оси от 0 до 1
```

Размер шрифта

Размер и другие свойства шрифта, который используется в `matplotlib` по умолчанию, можно изменить с помощью объекта `matplotlib.rcParams`:

```
matplotlib.rcParams.update({'font.size': 14})
```

Объект `matplotlib.rcParams` хранит множество настроек, изменяя которые, можно управлять поведением по умолчанию. Смотрите подробнее в [документации](#).

Подписи осей

Подписи к осям задаются следующим образом:

```
plt.xlabel('run number', fontsize=16)
plt.ylabel(r'average current ( $\mu$  A)', fontsize=16)
```

В подписях к осям (и вообще в любом тексте в matplotlib) можно использовать инструменты текстовой разметки [TeX](#), позволяющие отрисовывать различные математические выражения. TeX-выражения должны быть внутри пары символов $, кроме того, их следует помещать в r-строки, чтобы избежать неправильной обработки.$

Заголовок

Функция `pyplot.title` задает заголовок диаграммы. Применим наши новые знания:

```
import numpy as np
import matplotlib.pyplot as plt
import matplotlib

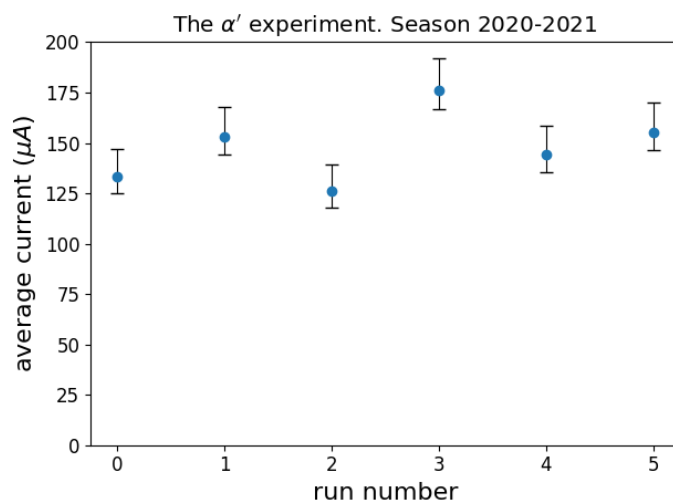
# задаем размер шрифта
matplotlib.rcParams.update({'font.size': 12})

rg = np.random.Generator(np.random.PCG64(11))
x = np.arange(6)
y = rg.poisson(149, x.size)
yerr = [
    0.7*np.sqrt(y),
    1.2*np.sqrt(y)
]
plt.errorbar(x, y, yerr=yerr, marker='o', linestyle='none',
             ecolor='k', elinewidth=0.8, capsize=4, capthick=1)

# добавляем подписи к осям и заголовок диаграммы
plt.xlabel('run number', fontsize=16)
plt.ylabel(r'average current ( $\mu$  A)', fontsize=16)
plt.title(r'The  $\alpha'$  experiment. Season 2020-2021')

# задаем диапазон значений оси y
plt.ylim([0, 200])
# оптимизируем поля и расположение объектов
plt.tight_layout()

plt.show()
```



В этом примере мы использовали функцию [pyplot.tight_layout](#), которая автоматически подбирает параметры отображения так, чтобы различные элементы не пересекались.

Легенда

При построении нескольких графиков в одних осях полезно добавлять легенду — пояснения к каждой линии. Следующий пример показывает, как это делается с помощью аргументов `label` и функции [pyplot.legend](#):

```
import numpy as np
import matplotlib.pyplot as plt
import matplotlib

matplotlib.rcParams.update({'font.size': 12})

x = np.linspace(0, 1, 100)
f1 = 0.25 - (x - 0.5)**2
f2 = x**3

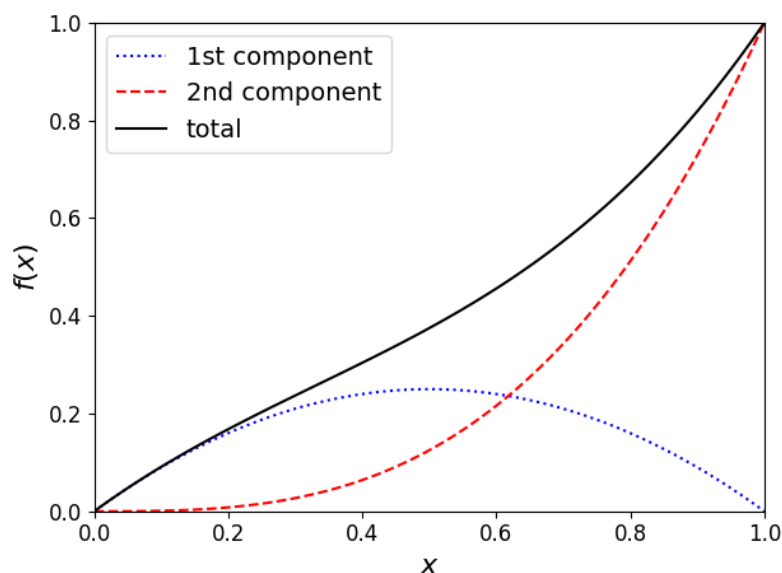
# указываем в аргументе label содержание легенды
plt.plot(x, f1, ':b', label='1st component')
plt.plot(x, f2, '--r', label='2nd component')
plt.plot(x, f1+f2, 'k', label='total')

plt.xlabel(r'$x$', fontsize=16)
plt.ylabel(r'$f(x)$', fontsize=16)

plt.xlim([0, 1])
plt.ylim([0, 1])

# выводим легенду
plt.legend(fontsize=14)

plt.tight_layout()
plt.show()
```



Функция `pyplot.legend` старается расположить легенду так, чтобы она не пересекала графики. Аргумент `loc` позволяет задать расположение легенды вручную. В большинстве случаев расположение по умолчанию получается удачным. Детали и описание других аргументов смотрите в [документации](#).

Сетка

Сетка во многих случаях облегчает анализ графиков. Включить отображение сетки можно с помощью функции `pyplot.grid`. Аргумент `axis` этой функции имеет три возможных значения: `x`, `y` и `both` и определяет оси, вдоль которых будут проведены линии сетки. Управлять свойствами линии сетки можно с помощью именованных аргументов, которые мы рассматривали выше при обсуждении функции `pyplot.plot`.

В `matplotlib` поддерживается два типа сеток: основная и дополнительная. Выбор типа сетки выполняется с помощью аргумента `which`, который может принимать три значения: `major`, `minor` и `both`. По умолчанию используется основная сетка.

Линии сетки привязаны к отметкам на осях. Чтобы работать с дополнительной сеткой необходимо сначала включить вспомогательные отметки на осях (которые по умолчанию отключены и к которым привязаны линии дополнительной сетки) с помощью функции `pyplot.minorticks_on`. Приведем пример:

```
import numpy as np
import matplotlib.pyplot as plt
import matplotlib

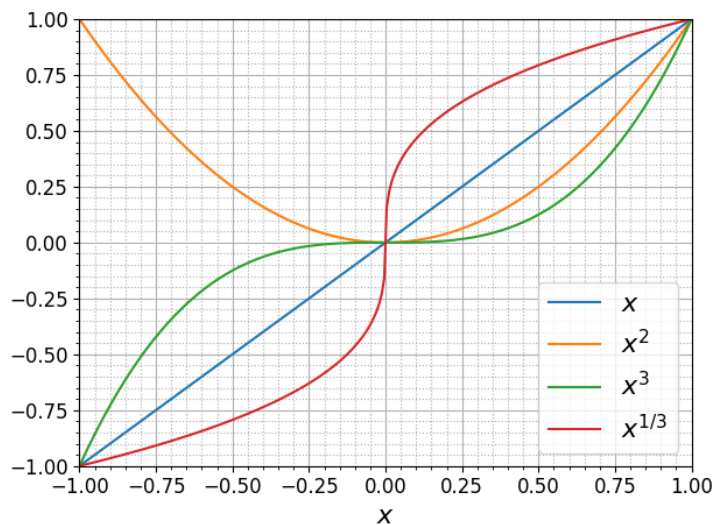
matplotlib.rcParams.update({'font.size': 12})

x = np.linspace(-1, 1, 250)
plt.plot(x, x, label=r'$x$')
plt.plot(x, x**2, label=r'$x^2$')
plt.plot(x, x**3, label=r'$x^3$')
plt.plot(x, np.cbrt(x), label=r'$x^{1/3}$')
plt.legend(fontsize=16)

# включаем дополнительные отметки на осях
plt.minorticks_on()
plt.xlabel(r'$x$', fontsize=16)

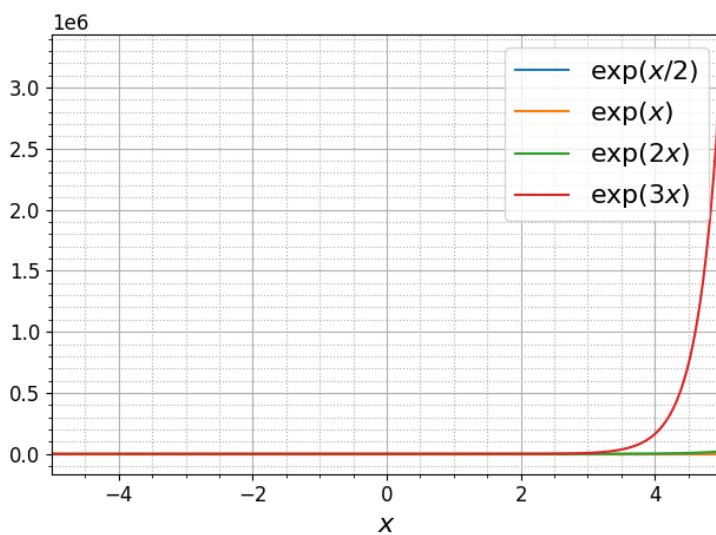
plt.xlim([-1., 1.])
plt.ylim([-1., 1.])
# включаем основную сетку
plt.grid(which='major')
# включаем дополнительную сетку
plt.grid(which='minor', linestyle=':')
plt.tight_layout()

plt.show()
```



Логарифмический масштаб

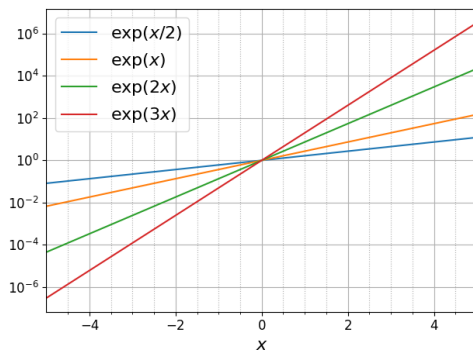
Функции `pyplot.semilogy` и `pyplot.semilogx` выполняют переключение между линейным и логарифмическим масштабами осей. В некоторых случаях логарифмический масштаб позволяет отобразить особенности зависимостей, которые не видны в линейном масштабе. Вот так выглядят графики экспоненциальных функций в линейном масштабе:



Добавление строки

```
plt.semilogy()
```

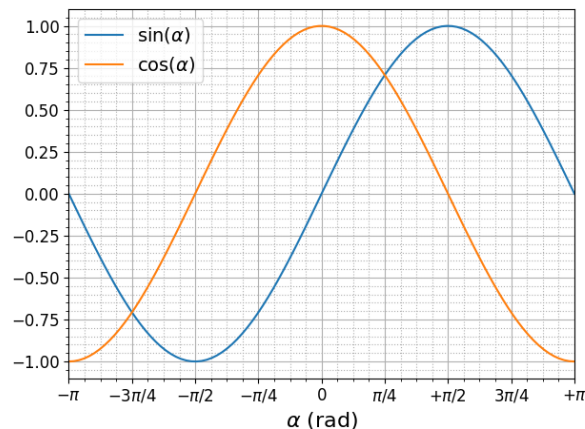
делает график гораздо более информативным:



Теперь мы видим поведение функций во всем динамическом диапазоне, занимающем 12 порядков.

Произвольные отметки на осях

Вернемся к первому примеру, в котором мы строили графики синуса и косинуса. Сделаем так, чтобы на горизонтальной оси отметки соответствовали различным долям числа π и имели соответствующие подписи:



Метки на горизонтальной оси были заданы с помощью функции `pyplot.xticks`:

```
plt.xticks(
    np.linspace(-np.pi, np.pi, 9),
    [r'$-\pi$', r'$-3\pi/4$', r'$-\pi/2$', r'$-\pi/4$', r'$0$',
     r'$\pi/4$', r'$+\pi/2$', r'$3\pi/4$', r'$+\pi$'])
```

Модуль `pyplot.ticker` содержит более продвинутые инструменты для управления отметками на осях. Подробности смотрите в [документации](#).

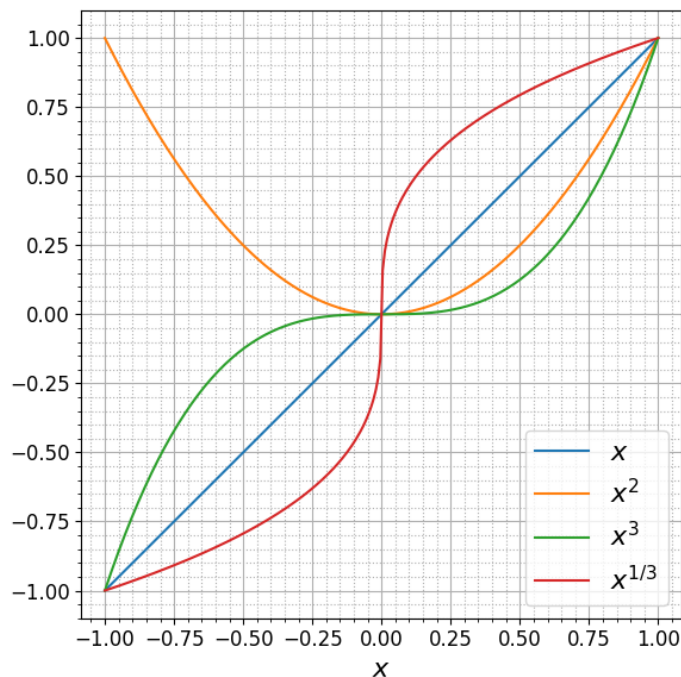
Размер изображения

До сих пор мы строили графики в одном окне, размер которого был задан по умолчанию. За кадром `matplotlib` создавал объект типа `Figure`, который определяет размер окна и содержит все остальные элементы. Кроме того, автоматически создавался объект типа `Axis`. Подробнее работа с этими объектами будет рассмотрена ниже. Сейчас же мы рассмотрим функцию `pyplot.figure`, которая позволяет создавать новые объекты типа `Figure` и переключаться между уже созданными объектами.

Функция `pyplot.figure` может принимать множество аргументов. Вот основные:

- `num: int` или `str` — уникальный идентификатор объекта типа. Если задан новый идентификатор, то создается новый объект и он становится активным. В случае, если передан идентификатор уже существующего объекта, то этот объект возвращается и становится активным
- `figsize: (float, float)` — размер изображения в дюймах
- `dpi: float` — разрешение в количестве точек на дюйм

Описание других параметров функции `pyplot.figure` можно найти в [документации](#). Используем эту функцию и функцию `pyplot.axis` чтобы улучшить наш пример с построением степенных функций:



Мы добавили две строки по сравнению с прошлой версией:

```
fig = plt.figure(figsize=(6, 6))
# ...
plt.axis('equal')
```

Функция `pyplot.axis` позволяет задавать некоторые свойства осей. Ее вызов с параметром `'equal'` делает одинаковыми масштабы вертикальной и горизонтальной осей, что кажется хорошей идеей в этом примере. Функция `pyplot.axis` возвращает кортеж из четырех значений `xmin`, `xmax`, `ymin`, `ymax`, соответствующих границам диапазонов значений осей.

Некоторые другие способы использования функции `pyplot.axis`:

- Кортеж из четырех `float` задаст новые границы диапазонов значений осей
- Строка `'off'` выключит отображение линий и меток осей

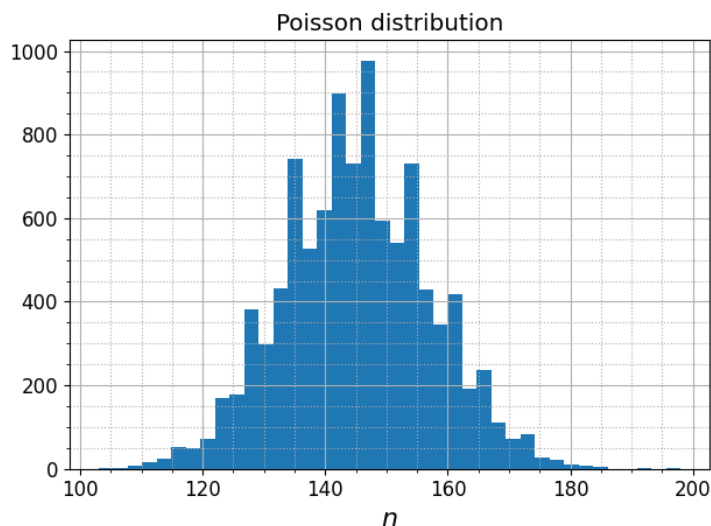
Гистограммы

Обратимся теперь к другим типам диаграмм. Функция `pyplot.hist` строит гистограмму по набору значений:

```
import numpy as np
```

```
import matplotlib.pyplot as plt

rg = np.random.Generator(np.random.PCG64(5))
data = rg.poisson(145, 10000)
plt.hist(data, bins=40)
# для краткости мы опускаем код для настройки осей, сетки и т.д.
```

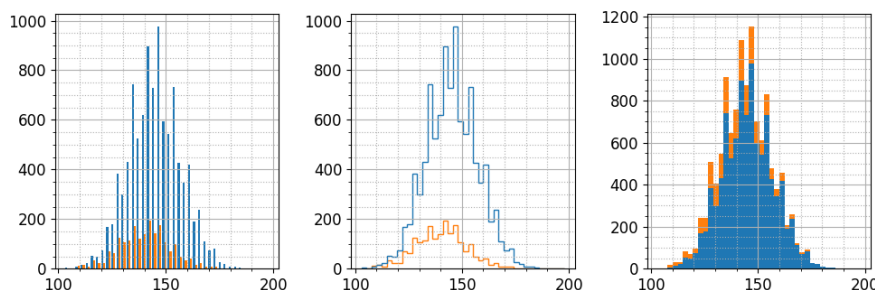


Аргумент `bins` задает количество бинов гистограммы. По умолчанию используется значение 10. Если вместо целого числа в аргумент `bins` передать кортеж значений, то они будут использованы для задания границ бинов. Таким образом можно построить гистограмму с произвольным разбиением.

Некоторые другие аргументы функции `pyplot.hist`:

- `range: (float, float)` — диапазон значений, в котором строится гистограмма. Значения за пределами заданного диапазона игнорируются.
- `density: bool`. При значении `True` будет построена гистограмма, соответствующая плотности вероятности, так что площадь гистограммы будет равна единице.
- `weights`: список `float` значений того же размера, что и набор данных. Определяет вес каждого значения при построении гистограммы.
- `histtype: str`. может принимать значения `{'bar', 'barstacked', 'step', 'stepfilled'}`. Определяет тип отрисовки гистограммы.

В качестве первого аргумента можно передать кортеж наборов значений. Для каждого из них будет построена гистограмма. Аргумент `stacked` со значением `True` позволяет строить сумму гистограмм для кортежа наборов. Покажем несколько примеров:



```
rg = np.random.Generator(np.random.PCG64(5))
```



```

data1 = rg.poisson(145, 10000)
data2 = rg.poisson(140, 2000)

# левая гистограмма
plt.hist([data1, data2], bins=40)
# центральная гистограмма
plt.hist([data1, data2], bins=40, histtype='step')
# правая гистограмма
plt.hist([data1, data2], bins=40, stacked=True)

```

В физике гистограммы часто представляют в виде набора значений с ошибками, предполагая при этом, что количество событий в каждом бине является случайной величиной, подчиняющейся биномиальному распределению. В пределе больших значений флуктуации количества событий в бине могут быть описаны распределением Пуассона, так что характерная величина флуктуации определяется корнем из числа событий. Библиотека **matplotlib** не имеет инструмента для такого представления данных, однако его легко получить с помощью комбинации **numpy.histogram** и **pyplot.errorbar**:

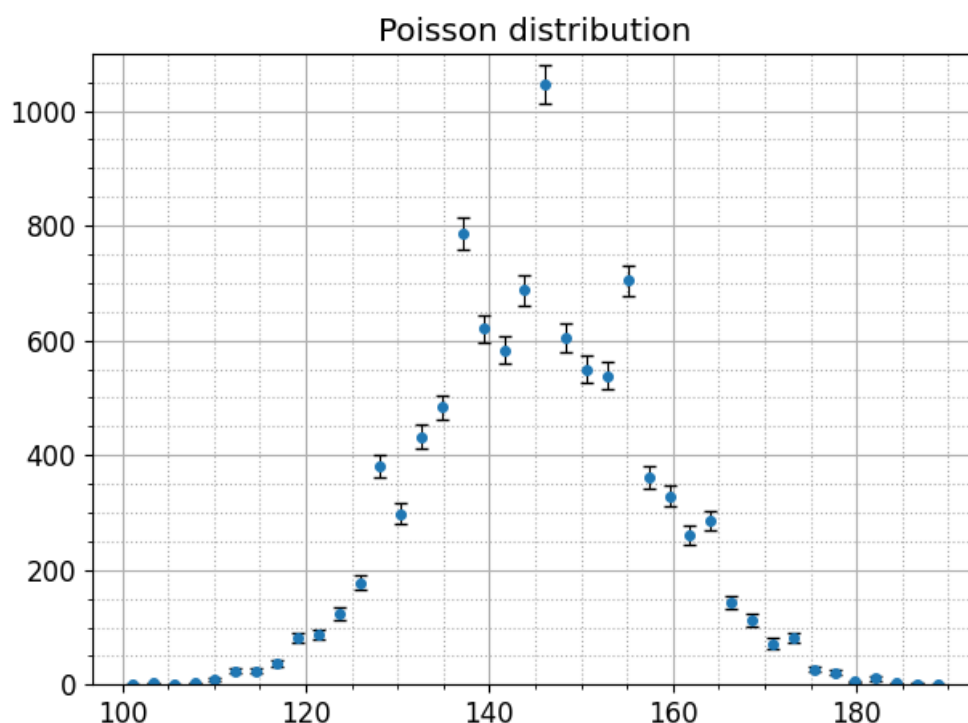
```

def poisson_hist(data, bins=60, lims=None):
    """ Гистограмма в виде набора значений с ошибками """
    hist, bins = np.histogram(data, bins=bins, range=lims)
    bins = 0.5 * (bins[1:] + bins[:-1])
    return (bins, hist, np.sqrt(hist))

rg = np.random.Generator(np.random.PCG64(5))
data = rg.poisson(145, 10000)

x, y, yerr = poisson_hist(data, bins=40, lims=(100, 190))
plt.errorbar(x, y, yerr=yerr, marker='o', markersize=4,
             linestyle='none', ecolor='k', elinewidth=0.8,
             capsize=3, capthick=1)

```



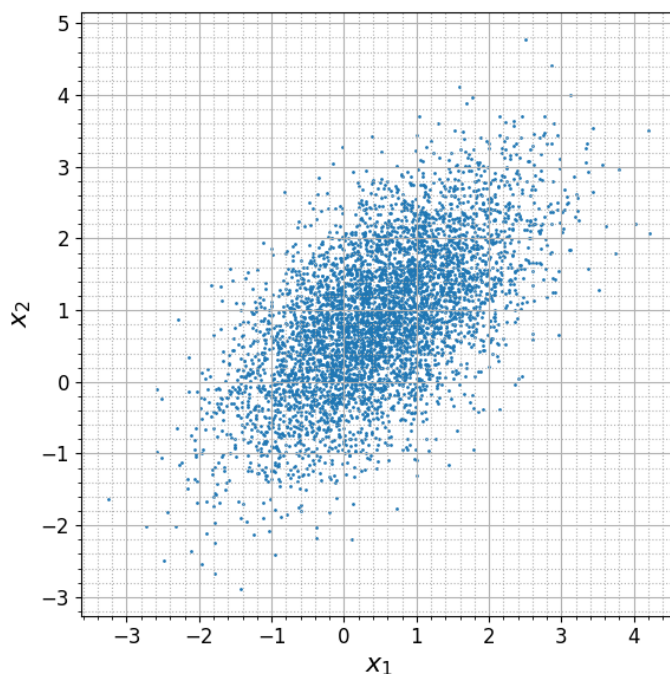
Диаграммы рассеяния

Распределение событий по двум измерениям удобно визуализировать с помощью [диаграммы рассеяния](#):

```
rg = np.random.Generator(np.random.PCG64(5))

means = (0.5, 0.9)
covar = [
    [1., 0.6],
    [0.6, 1.]
]
data = rg.multivariate_normal(means, covar, 5000)

plt.scatter(data[:,0], data[:,1], marker='o', s=1)
```

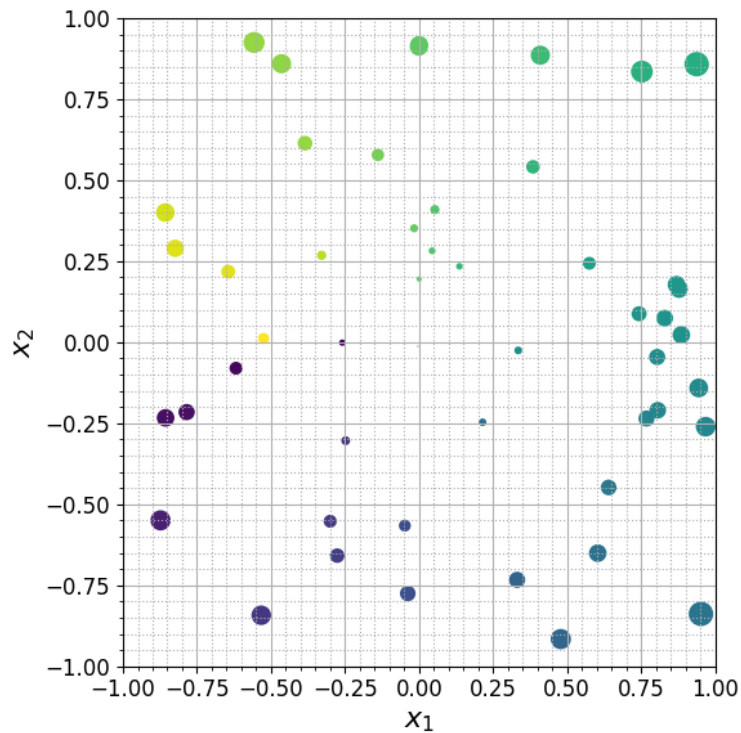


Каждой паре значений в наборе данных соответствует одна точка на диаграмме. Несмотря на свою простоту, диаграмма рассеяния позволяет во многих случаях наглядно представлять двумерные данные. Функция [pyplot.scatter](#) позволяет визуализировать и данные более высокой размерности: размер и цвет маркера могут быть заданы для каждой точки отдельно:

```
rg = np.random.Generator(np.random.PCG64(4))

data = rg.uniform(-1, 1, (50, 2))
col = np.arctan2(data[:, 1], data[:, 0])
size = 100*np.sum(data**2, axis=1)

plt.scatter(data[:,0], data[:,1], marker='o', s=size, c=col)
```



Цветовую палитру можно задать с помощью аргумента `cmap`. Подробности и описание других аргументов функции `pyplot.scatter` можно найти в [документации](#).

Контурные диаграммы

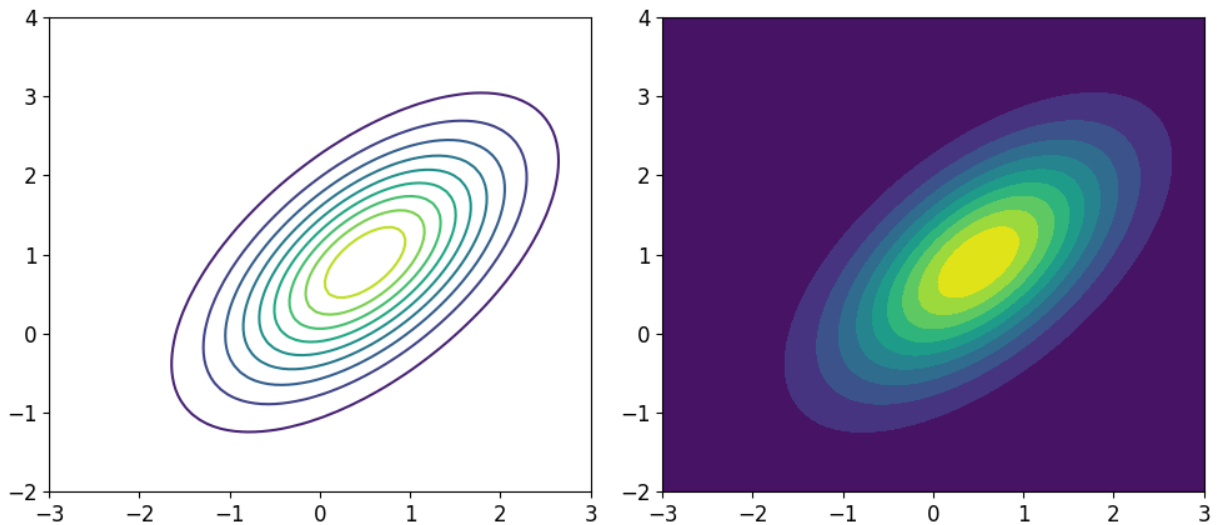
Контурные диаграммы позволяют визуализировать функции двух переменных:

```
from scipy import stats

means = (0.5, 0.9)
covar = [
    [1., 0.6],
    [0.6, 1.]
]

mvn = stats.multivariate_normal(means, covar)
x, y = np.meshgrid(
    np.linspace(-3, 3, 80),
    np.linspace(-2, 4, 80)
)
data = np.dstack((x, y))

# левая диаграмма — без заливки цветом
plt.contour(x, y, mvn.pdf(data), levels=10)
# правая диаграмма — с заливкой цветом
plt.contourf(x, y, mvn.pdf(data), levels=10)
```

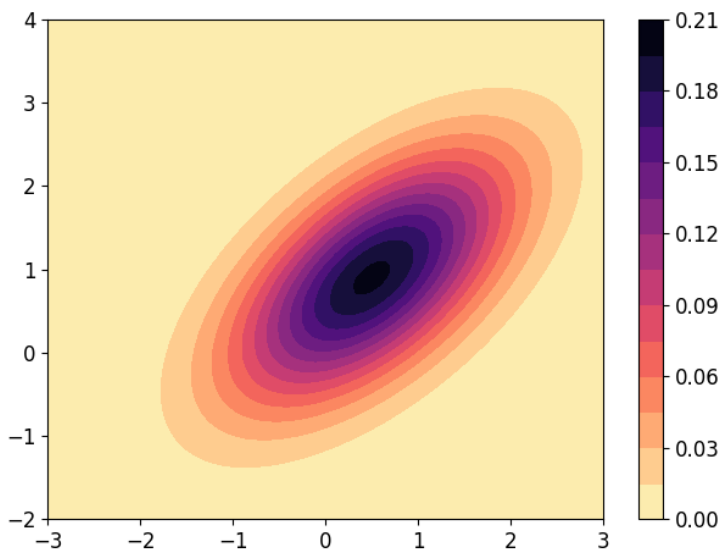


Аргумент `levels` задает количество контуров. По умолчанию контуры отрисовываются равномерно между максимальным и минимальным значениями. В аргумент `levels` также можно передать список уровней, на которых следует провести контуры.

Обратите внимание на использование функций [numpy.meshgrid](#) и [numpy.dstack](#) в этом примере.

Контурную диаграмму можно дополнить цветовой полосой `colorbar`, вызвав функцию `pyplot.colorbar`:

```
cs = plt.contourf(x, y, mvn.pdf(data), levels=15,
                  cmap=matplotlib.cm.magma_r)
cbar = plt.colorbar(cs)
```



Более подробное описание функций `plt.contour` и `plt.contourf` смотрите в [документации](#).

Расположение нескольких осей в одном окне

В одном окне (объекте `Figure`) можно разместить несколько осей (объектов `axis.Axis`). Функция `pyplot.subplots` создает объект `Figure`, содержащий регулярную сетку объектов `axis.Axis`:

```

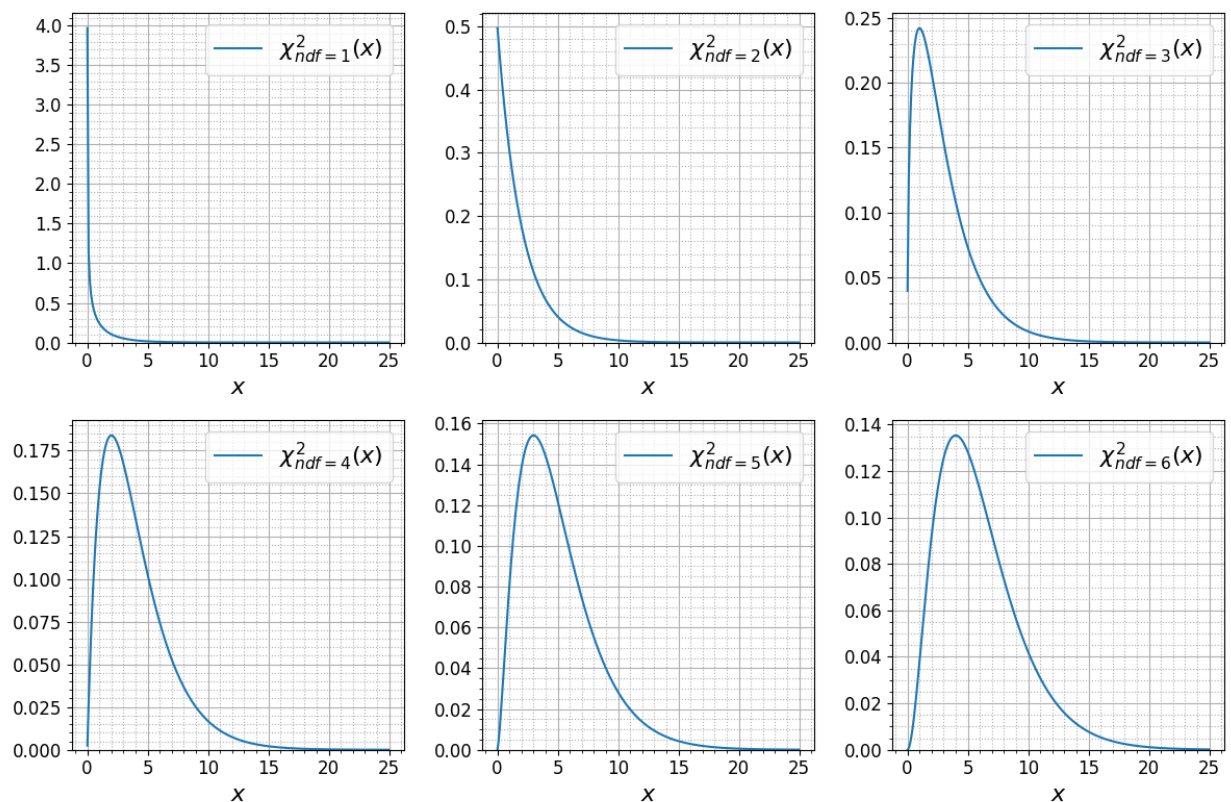
import numpy as np
from scipy import stats
import matplotlib.pyplot as plt

fig, axes = plt.subplots(ncols=3, nrows=2, figsize=(12, 8))
x = np.linspace(0.01, 25, 250)

for idx, row in enumerate(axes):
    for jdx, ax in enumerate(row):
        ndf = idx * 3 + jdx + 1
        y = stats.chi2.pdf(x, ndf)
        ax.plot(x, y, label=fr'$\chi^2_{ndf={ndf}}(x)$')
        ax.set_xlabel(r'$x$', fontsize=16)
        ax.set_ylim([0, 1.05*y.max()])
        ax.minorticks_on()
        ax.legend(fontsize=16)
        ax.grid(which='major')
        ax.grid(which='minor', linestyle=':')

fig.tight_layout()
plt.show()

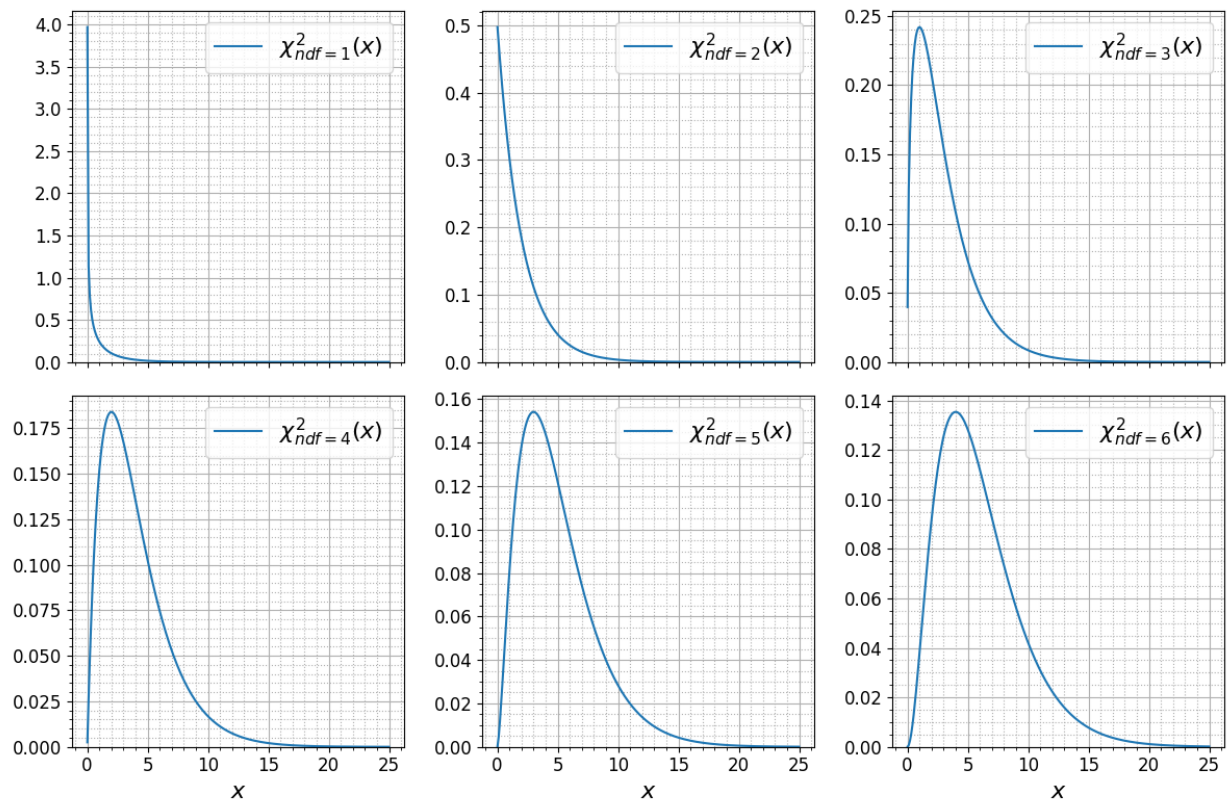
```



Количество строк и столбцов, по которым располагаются различные оси, задаются с помощью параметров `nrows` и `ncols`, соответственно. Функция `pyplot.subplots` возвращает объект `Figure` и двумерный список осей `axis.Axis`. Обратите внимание на то, что вместо вызовов функций модуля `pyplot` в этом примере использовались вызовы методов классов `Figure` и `axis.Axis`.

В последнем примере горизонтальная ось во всех графиках имеет один и тот же диапазон. Аргумент `sharex` функции `pyplot.subplots` позволяет убрать дублирование отрисовки осей в таких случаях:

```
fig, axes = plt.subplots(ncols=3, nrows=2, figsize=(12, 8),
                          sharex=True)
# ...
for idx, row in enumerate(axes):
    for jdx, ax in enumerate(row):
        # ...
        if idx:
            ax.set_xlabel(r'$x$', fontsize=16)
```



Существует аналогичный параметр `sharey` для вертикальной оси.

Более гибкие возможности регулярного расположения осей предоставляет функция `pyplot.subplot`. Мы не будем рассматривать эту функцию и ограничимся лишь ее упоминанием.

Функция `pyplot.axes` позволяет добавлять новые оси в текущем окне в произвольном месте:

```
import numpy as np
import matplotlib.pyplot as plt

exno = 26

rg = np.random.Generator(np.random.PCG64(5))
x1 = rg.exponential(10, 5000)
x2 = rg.normal(10, 0.1, 100)
```

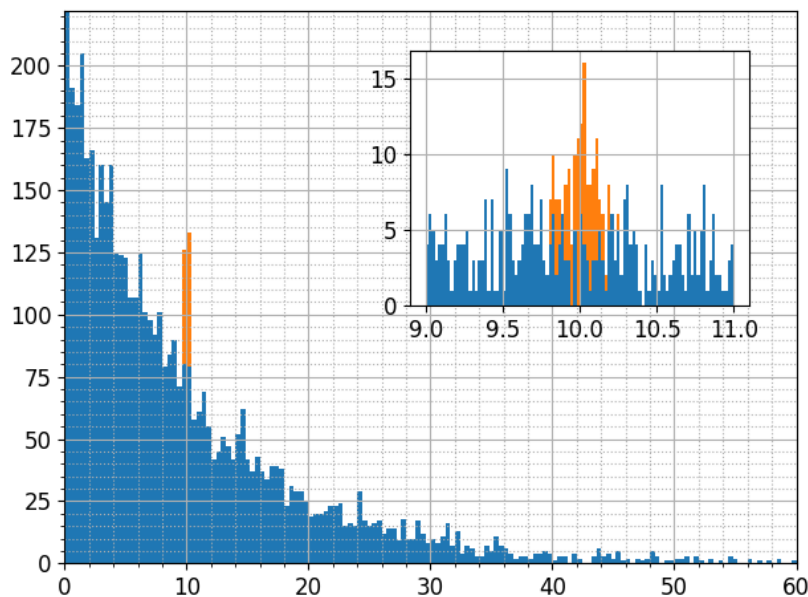


```
# Строим основную гистограмму
plt.hist([x1, x2], bins=150, range=(0, 60), stacked=True)
plt.minorticks_on()
plt.xlim((0, 60))
plt.grid(which='major')
plt.grid(which='minor', linestyle=':')

# Строим вторую гистограмму в отдельных осях
plt.axes([.5, .5, .4, .4])
plt.hist([x1, x2], bins=100, stacked=True, range=(9, 11))
plt.grid(which='major')

plt.tight_layout()

# сохраняем диаграмму в файл
plt.savefig('histograms.png')
plt.show()
```



В этом примере была использована функция `pyplot.savefig`, сохраняющая содержимое текущего окна в файл в векторном или растровом формате. Формат задается с помощью аргумента `format` или автоматически определяется из имени файла (как в примере выше). Набор доступных форматов зависит от окружения, однако в большинстве случаев можно использовать такие форматы как `png`, `jpeg`, `pdf`, `svg` и `eps`.

Резюме

Предметом изучения в этом разделе был модуль `pyplot` библиотеки `matplotlib`, содержащий инструменты для построения различных диаграмм. Были рассмотрены:

- функции `pyplot.plot`, `pyplot.errorbar`, `pyplot.hist`, `pyplot.scatter`, `pyplot.contour` и `pyplot.contourf` для построения диаграмм;
- средства настройки свойств линий и маркеров;
- средства настройки координатных осей: подписи, размер шрифта, координатная сетка, произвольные метки др.;

- инструменты для расположения нескольких координатных осей в одном окне.

Задания для выполнения.

Вариант 1

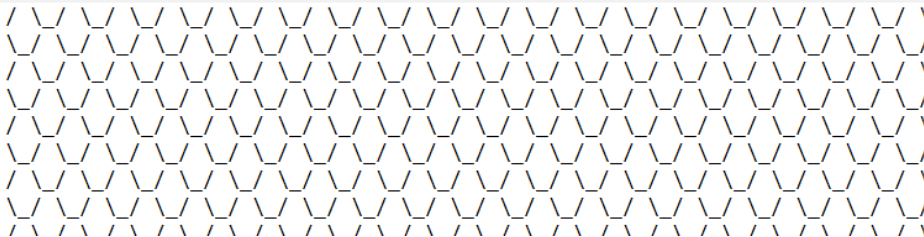
1. Напишите функцию, которая будет суммировать все элементы в списке.
2. Напишите функцию, которая будет записывать строку задом наперед, и записывать полученный результат в файл. Каждый запуск программы должен дозаписывать файл.
3. Напишите функцию для решения системы уравнений методом Крамера. Результат сохраняйте в файл. Вывести сообщение, если система уравнений не имеет решения. Система уравнений передается в функцию как три отдельных уравнения.

$$\begin{cases} 2x - 4y + 11z = 8 \\ -x + 2y - z = 9 \\ 4x + 3y + z = 7 \end{cases}$$

4. Напишите функцию, которая будет строить графики в одном окне. Используйте линии разного цвета, постройте легенду. В качестве функций используйте уравнения из задания 3. z – параметр, который пользователь вводит с клавиатуры.
5. Напишите функцию, которая будет генерировать и сохранять в текстовом документе мозаичное изображение как показано на рисунке

*Безымянный – Блокнот

Файл Правка Формат Вид Справка



6. Напишите функцию, которая будет генерировать случайный пароль. В пароле должно быть от 8 до 15 символов Юникода из диапазонов 48-57 (цифры), 65-90 (буквы латинского алфавита в верхнем регистре) и 97-122 (буквы латинского алфавита в нижнем регистре). Сгенерируйте и выведите на экран три пароля.
7. Напишите функцию, которая будет строить круговую диаграмму. Постройте с помощью этой функции круговую диаграмму состава вселенной
8. Напишите функцию для построения гистограммы, причем цвет столбцов



гистограммы должен градуироваться от максимального до минимального как .

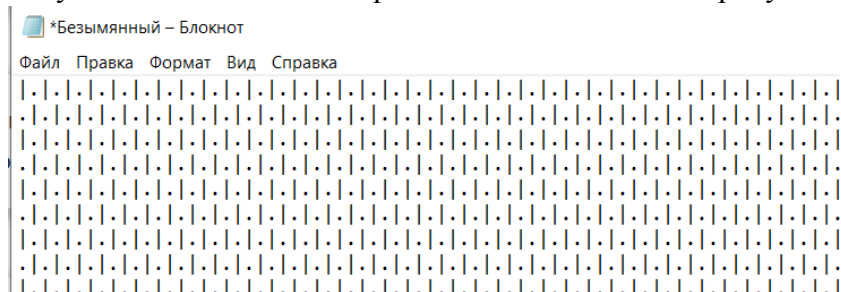
Напишите функцию, которая будет формировать список распределения частоты встречаемости различных букв в русскоязычном тексте художественной литературы, объемом не менее 10000 символов. Постройте гистограмму. Сохраните как рисунок .jpg

Вариант 2

1. Напишите функцию, которая будет находить произведение всех элементов в списке.
2. Напишите функцию, которая будет удалять из строки все гласные буквы, и записывать полученный результат в файл. Каждый запуск программы должен дозаписывать файл.
3. Напишите функцию для решения системы уравнений методом Крамера. Результат сохраняйте в файл. Вывести сообщение, если система уравнений не имеет решения. Система уравнений передается в функцию как три отдельных уравнения.

$$\begin{cases} 3x - 3y + 6z = 4 \\ 2x - y + 4z = 2 \\ 4x + 3y + z = 7 \end{cases}$$

4. Напишите функцию, которая будет строить графики в одном окне. Используйте линии разного цвета, постройте легенду. В качестве функций используйте уравнения из задания 3. z – параметр, который пользователь вводит с клавиатуры.
5. Напишите функцию, которая будет генерировать и сохранять в текстовом документе мозаичное изображение как показано на рисунке



6. Напишите функцию, которая будет генерировать случайный пароль. В пароле должно быть от 8 до 15 символов Юникода из диапазонов 48-57 (цифры), 65-90 (буквы латинского алфавита в верхнем регистре) и 97-122 (буквы латинского алфавита в нижнем регистре). Сгенерируйте и выведите на экран три пароля.
7. Напишите функцию, которая будет строить круговую диаграмму. Постройте с помощью этой функции круговую диаграмму состава морской воды
8. Напишите функцию для построения гистограммы, причем цвет столбцов

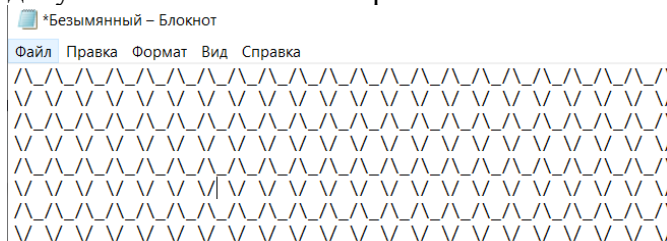
гистограммы должен градуироваться от максимального до минимального как .
 Напишите функцию, которая будет формировать список распределения частоты встречаемости различных букв в русскоязычном техническом тексте художественной литературы, объемом не менее 10000 символов. Постройте гистограмму. Сохраните как рисунок .jpg

Вариант 3

1. Напишите функцию, которая будет находить максимальный элемент в списке.
2. Напишите функцию, которая будет удалять из строки все согласные буквы, и записывать полученный результат в файл. Каждый запуск программы должен дозаписывать файл.
3. Напишите функцию для решения системы уравнений методом Крамера. Результат сохраняйте в файл. Вывести сообщение, если система уравнений не имеет решения. Система уравнений передается в функцию как три отдельных уравнения.

$$\begin{cases} 3x - 3y - 2z = 3 \\ 4x - 2y + z = -2 \\ 4x + 3y + z = 7 \end{cases}$$

4. Напишите функцию, которая будет строить графики в одном окне. Используйте линии разного цвета, постройте легенду. В качестве функций используйте уравнения из задания 3. z – параметр, который пользователь вводит с клавиатуры.
5. Напишите функцию, которая будет генерировать и сохранять в текстовом документе мозаичное изображение как показано на рисунке



6. Напишите функцию, которая будет генерировать случайный пароль. В пароле должно быть от 8 до 15 символов Юникода из диапазонов 48-57 (цифры), 65-90 (буквы латинского алфавита в верхнем регистре) и 97-122 (буквы латинского алфавита в нижнем регистре). Сгенерируйте и выведите на экран три пароля.
7. Напишите функцию, которая будет строить круговую диаграмму. Постройте с помощью этой функции круговую диаграмму состава воздуха
8. Напишите функцию для построения гистограммы, причем цвет столбцов

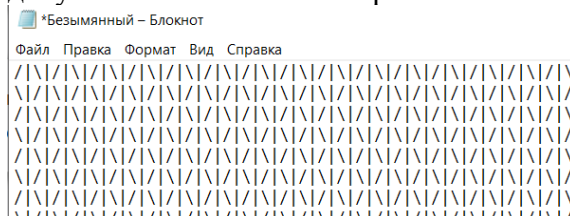
гистограммы должен градуироваться от максимального до минимального как .
 Напишите функцию, которая будет формировать список распределения частоты встречаемости различных букв в англоязычном тексте художественной литературы, объемом не менее 10000 символов. Постройте гистограмму. Сохраните как рисунок .jpg

Вариант 4


1. Напишите функцию, которая будет находить минимальный элемент в списке
2. Напишите функцию, которая будет заменять все гласные буквы на букву A, и записывать полученный результат в файл. Каждый запуск программы должен дозаписывать файл.
3. Напишите функцию для решения системы уравнений методом Крамера. Результат сохраняйте в файл. Вывести сообщение, если система уравнений не имеет решения. Система уравнений передается в функцию как три отдельных уравнения.

$$\begin{cases} 4x - 2y + 7z = 6 \\ -2x - y - 2z = 4 \\ 7x + 6y - z = 7 \end{cases}$$

4. Напишите функцию, которая будет строить графики в одном окне. Используйте линии разного цвета, постройте легенду. В качестве функций используйте уравнения из задания 3. z – параметр, который пользователь вводит с клавиатуры.
5. Напишите функцию, которая будет генерировать и сохранять в текстовом документе мозаичное изображение как показано на рисунке



6. Напишите функцию, которая будет генерировать случайный пароль. В пароле должно быть от 8 до 15 символов Юникода из диапазонов 48-57 (цифры), 65-90 (буквы латинского алфавита в верхнем регистре) и 97-122 (буквы латинского алфавита в нижнем регистре). Сгенерируйте и выведите на экран три пароля.
7. Напишите функцию, которая будет строить круговую диаграмму. Постройте с помощью этой функции круговую диаграмму количества железа в различных продуктах
8. Напишите функцию для построения гистограммы, причем цвет столбцов

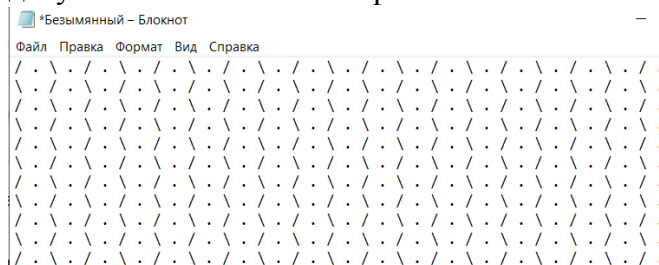
гистограммы должен градуироваться от максимального до минимального как . Напишите функцию, которая будет формировать список распределения частоты встречаемости различных букв в англоязычном техническом тексте, объемом не менее 10000 символов. Постройте гистограмму. Сохраните как рисунок .jpg

Вариант 5

1. Напишите функцию, которая будет суммировать все элементы в списке.
2. Напишите функцию, которая будет заменять все пробелы в строке на нижнее подчеркивание, и записывать полученный результат в файл. Каждый запуск программы должен дозаписывать файл.
3. Напишите функцию для решения системы уравнений методом Крамера. Результат сохраняйте в файл. Вывести сообщение, если система уравнений не имеет решения. Система уравнений передается в функцию как три отдельных уравнения.

$$\begin{cases} -x - 3y - 3z = -8 \\ 2x + 4y + z = 4 \\ x - 3y + z = 3 \end{cases}$$

4. Напишите функцию, которая будет строить графики в одном окне. Используйте линии разного цвета, постройте легенду. В качестве функций используйте уравнения из задания 3. z – параметр, который пользователь вводит с клавиатуры.
5. Напишите функцию, которая будет генерировать и сохранять в текстовом документе мозаичное изображение как показано на рисунке



6. Напишите функцию, которая будет генерировать случайный пароль. В пароле должно быть от 8 до 15 символов Юникода из диапазонов 48-57 (цифры), 65-90 (буквы латинского алфавита в верхнем регистре) и 97-122 (буквы латинского алфавита в нижнем регистре). Сгенерируйте и выведите на экран три пароля.
7. Напишите функцию, которая будет строить круговую диаграмму. Постройте с помощью этой функции круговую диаграмму микроэлементов в составе огурца

8. Напишите функцию для построения гистограммы, причем цвет столбцов



гистограммы должен градуироваться от максимального до минимального как . Напишите функцию, которая будет формировать список распределения частоты встречаемости различных букв в русскоязычном тексте художественной литературы, объемом не менее 10000 символов. Постройте гистограмму. Сохраните как рисунок .jpg

Вариант 6

1. Напишите функцию, которая будет находить произведение всех элементов в списке.
2. Напишите функцию, которая будет удалять все пробелы в строке, и записывать полученный результат в файл. Каждый запуск программы должен дозаписывать файл.
3. Напишите функцию для решения системы уравнений методом Крамера. Результат сохраняйте в файл. Вывести сообщение, если система уравнений не имеет решения. Система уравнений передается в функцию как три отдельных уравнения.

$$\begin{cases} 3x + 3y - z = -8 \\ -2x - 2y + 4z = 3 \\ 7x + 2y - z = 7 \end{cases}$$

4. Напишите функцию, которая будет строить графики в одном окне. Используйте линии разного цвета, постройте легенду. В качестве функций используйте уравнения из задания 3. z – параметр, который пользователь вводит с клавиатуры.
5. Напишите функцию, которая будет генерировать и сохранять в текстовом документе мозаичное изображение как показано на рисунке



6. Напишите функцию, которая будет генерировать случайный пароль. В пароле должно быть от 8 до 15 символов Юникода из диапазонов 48-57 (цифры), 65-90 (буквы латинского алфавита в верхнем регистре) и 97-122 (буквы латинского алфавита в нижнем регистре). Сгенерируйте и выведите на экран три пароля.

7. Напишите функцию, которая будет строить круговую диаграмму. Постройте с помощью этой функции круговую диаграмму содержания витамина С в различных продуктах
8. Напишите функцию для построения гистограммы, причем цвет столбцов

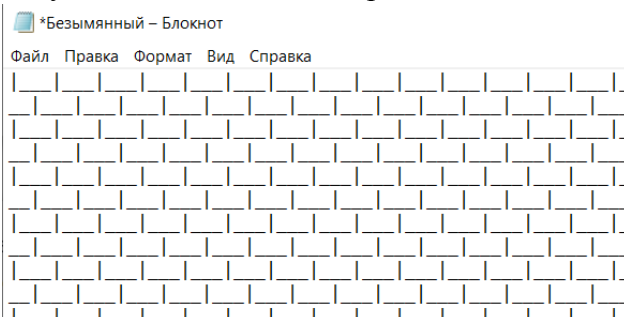


гистограммы должен градуироваться от максимального до минимального как . Напишите функцию, которая будет формировать список распределения частоты встречаемости различных букв в русскоязычном техническом тексте художественной литературы, объемом не менее 10000 символов. Постройте гистограмму. Сохраните как рисунок .jpg

Вариант 7

1. Напишите функцию, которая будет суммировать все элементы в списке.
2. Напишите функцию, которая заменит в каждом слове первую букву на заглавную, и записывать полученный результат в файл. Каждый запуск программы должен дозаписывать файл.
3. Напишите функцию для решения системы уравнений методом Крамера. Результат сохраняйте в файл. Вывести сообщение, если система уравнений не имеет решения. Система уравнений передается в функцию как три отдельных уравнения.

$$\begin{cases} -3x + 7y - 5z = 5 \\ 3x - 4y + 2z = 0 \\ 9x - y + 3z = 1 \end{cases}$$
4. Напишите функцию, которая будет строить графики в одном окне. Используйте линии разного цвета, постройте легенду. В качестве функций используйте уравнения из задания 3. z – параметр, который пользователь вводит с клавиатуры.
5. Напишите функцию, которая будет генерировать и сохранять в текстовом документе мозаичное изображение как показано на рисунке\



6. Напишите функцию, которая будет генерировать случайный пароль. В пароле должно быть от 8 до 15 символов Юникода из диапазонов 48-57 (цифры), 65-90 (буквы латинского алфавита в верхнем регистре) и 97-122 (буквы латинского алфавита в нижнем регистре). Сгенерируйте и выведите на экран три пароля.
7. Напишите функцию, которая будет строить круговую диаграмму. Постройте с помощью этой функции круговую диаграмму состава золотого кольца 585 пробы
8. Напишите функцию для построения гистограммы, причем цвет столбцов



гистограммы должен градуироваться от максимального до минимального как . Напишите функцию, которая будет формировать список распределения частоты встречаемости различных букв в англоязычном тексте художественной литературы, объемом не менее 10000 символов. Постройте гистограмму. Сохраните как рисунок .jpg

Вариант 8

1. Напишите функцию, которая будет находить максимальный элемент в списке.
2. Напишите функцию, которая будет добавлять к каждому слову строки порядковый номер, и записывать полученный результат в файл. Каждый запуск программы должен дозаписывать файл.
3. Напишите функцию для решения системы уравнений методом Крамера. Результат сохраняйте в файл. Вывести сообщение, если система уравнений не имеет решения. Система уравнений передается в функцию как три отдельных уравнения.

$$\begin{cases} -x + 3y - 3z = 4 \\ 10x - 2y - 2z = 3 \\ 2x - 4y + 2z = -6 \end{cases}$$

4. Напишите функцию, которая будет строить графики в одном окне. Используйте линии разного цвета, постройте легенду. В качестве функций используйте уравнения из задания 3. z – параметр, который пользователь вводит с клавиатуры.
5. Напишите функцию, которая будет генерировать и сохранять в текстовом документе мозаичное изображение как показано на рисунке



*Безымянный – Блокнот

Файл Правка Формат Вид Справка



6. Напишите функцию, которая будет генерировать случайный пароль. В пароле должно быть от 8 до 15 символов Юникода из диапазонов 48-57 (цифры), 65-90 (буквы латинского алфавита в верхнем регистре) и 97-122 (буквы латинского алфавита в нижнем регистре). Сгенерируйте и выведите на экран три пароля.
7. Напишите функцию, которая будет строить круговую диаграмму. Постройте с помощью этой функции круговую диаграмму содержания витаминов и микроэлементов в семенах подсолнуха
8. Напишите функцию для построения гистограммы, причем цвет столбцов

гистограммы должен градуироваться от максимального до минимального как

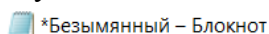
Напишите функцию, которая будет формировать список распределения частоты встречаемости различных букв в русскоязычном тексте художественной литературы, объемом не менее 10000 символов. Постройте гистограмму. Сохраните как рисунок .jpg

Вариант 9

1. Напишите функцию, которая будет находить минимальный элемент в списке.
2. Напишите функцию, которая будет менять местами каждые два соседних символа, и записывать полученный результат в файл. Каждый запуск программы должен дозаписывать файл.
3. Напишите функцию для решения системы уравнений методом Крамера. Результат сохраняйте в файл. Вывести сообщение, если система уравнений не имеет решения. Система уравнений передается в функцию как три отдельных уравнения.

$$\begin{cases} 3x + 2y - z = 7 \\ -2x + 4y - 2z = 4 \\ 3x - 5y + 3z = 1 \end{cases}$$

4. Напишите функцию, которая будет строить графики в одном окне. Используйте линии разного цвета, постройте легенду. В качестве функций используйте уравнения из задания 3. z – параметр, который пользователь вводит с клавиатуры.
5. Напишите функцию, которая будет генерировать и сохранять в текстовом документе мозаичное изображение как показано на рисунке



Файл Правка Формат Вид Справка

[illegible]

6. Напишите функцию, которая будет генерировать случайный пароль. В пароле должно быть от 8 до 15 символов Юникода из диапазонов 48-57 (цифры), 65-90 (буквы латинского алфавита в верхнем регистре) и 97-122 (буквы латинского алфавита в нижнем регистре). Сгенерируйте и выведите на экран три пароля
7. Напишите функцию, которая будет строить круговую диаграмму. Постройте с помощью этой функции круговую диаграмму содержания микроэлементов в банане.
8. Напишите функцию для построения гистограммы, причем цвет столбцов



гистограммы должен градуироваться от максимального до минимального как . Напишите функцию, которая будет формировать список распределения частоты встречаемости различных букв в англоязычном техническом тексте, объемом не менее 10000 символов. Постройте гистограмму. Сохраните как рисунок .jpg

Вариант 10

1. Напишите функцию, которая будет находить произведение всех элементов в списке.
2. Напишите функцию, которая будет удалять все знаки препинания в строке, и записывать полученный результат в файл. Каждый запуск программы должен дозаписывать файл.
3. Напишите функцию для решения системы уравнений методом Крамера. Результат сохраняйте в файл. Вывести сообщение, если система уравнений не имеет решения. Система уравнений передается в функцию как три отдельных уравнения.

$$\begin{cases} 3x + 2y - 4z = 3 \\ 6x - 2y - 4z = 2 \\ 4x + 3y + z = 7 \end{cases}$$
4. Напишите функцию, которая будет строить графики в одном окне. Используйте линии разного цвета, постройте легенду. В качестве функций используйте уравнения из задания 3. z – параметр, который пользователь вводит с клавиатуры.
5. Напишите функцию, которая будет генерировать и сохранять в текстовом документе мозаичное изображение как показано на рисунке



6. Напишите функцию, которая будет генерировать случайный пароль. В пароле должно быть от 8 до 15 символов Юникода из диапазонов 48-57 (цифры), 65-90 (буквы латинского алфавита в верхнем регистре) и 97-122 (буквы латинского алфавита в нижнем регистре). Сгенерируйте и выведите на экран три пароля.
7. Напишите функцию, которая будет строить круговую диаграмму. Постройте с помощью этой функции круговую диаграмму распределения цифр в последовательности простых чисел от 2 до 101.
8. Напишите функцию для построения гистограммы, причем цвет столбцов



гистограммы должен градуироваться от максимального до минимального как . Напишите функцию, которая будет формировать список распределения частоты встречаемости различных букв в англоязычном тексте художественной литературы, объемом не менее 10000 символов. Постройте гистограмму. Сохраните как рисунок .jpg

Вариант 11

1. Напишите функцию, которая будет находить максимальный элемент в списке.
2. Напишите функцию, которая будет записывать строку задом наперед, и записывать полученный результат в файл. Каждый запуск программы должен дозаписывать файл.
3. Напишите функцию для решения системы уравнений методом Крамера. Результат сохраняйте в файл. Вывести сообщение, если система уравнений не имеет решения. Система уравнений передается в функцию как три отдельных уравнения.

$$\begin{cases} 2x - 5y - 4z = 3 \\ -3x + 3y - 5z = 9 \\ 4x - y + 2z = 4 \end{cases}$$
4. Напишите функцию, которая будет строить графики в одном окне. Используйте линии разного цвета, постройте легенду. В качестве функций используйте уравнения из задания 3. z – параметр, который пользователь вводит с клавиатуры.

5. Напишите функцию, которая будет генерировать и сохранять в текстовом документе мозаичное изображение как показано на рисунке



6. Напишите функцию, которая будет генерировать случайный пароль. В пароле должно быть от 8 до 15 символов Юникода из диапазонов 48-57 (цифры), 65-90 (буквы латинского алфавита в верхнем регистре) и 97-122 (буквы латинского алфавита в нижнем регистре). Сгенерируйте и выведите на экран три пароля.
7. Напишите функцию, которая будет строить круговую диаграмму. Постройте с помощью этой функции круговую диаграмму количества металлов в земной коре
8. Напишите функцию для построения гистограммы, причем цвет столбцов



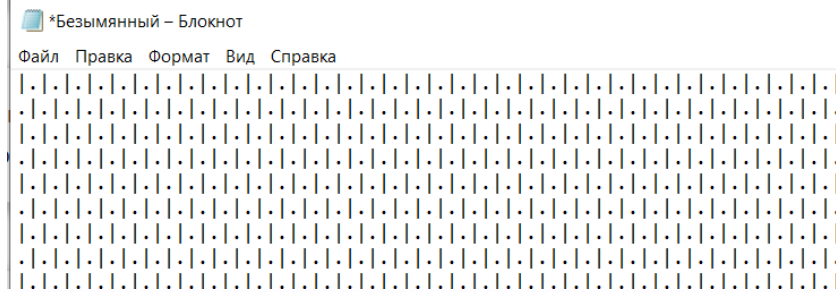
гистограммы должен градуироваться от максимального до минимального как . Напишите функцию, которая будет формировать список распределения частоты встречаемости различных букв в русскоязычном тексте художественной литературы, объемом не менее 10000 символов. Постройте гистограмму. Сохраните как рисунок .jpg

Вариант 12

1. Напишите функцию, которая будет находить минимальный элемент в списке
2. Напишите функцию, которая будет удалять из строки все гласные буквы, и записывать полученный результат в файл. Каждый запуск программы должен дозаписывать файл.
3. Напишите функцию для решения системы уравнений методом Крамера. Результат сохраняйте в файл. Вывести сообщение, если система уравнений не имеет решения. Система уравнений передается в функцию как три отдельных уравнения.

$$\begin{cases} 2x + 2y - 3z = 9 \\ -3x - 2y + 2z = 9 \\ 2x + 3y + 3z = -7 \end{cases}$$
4. Напишите функцию, которая будет строить графики в одном окне. Используйте линии разного цвета, постройте легенду. В качестве функций используйте уравнения из задания 3. z – параметр, который пользователь вводит с клавиатуры.

5. Напишите функцию, которая будет генерировать и сохранять в текстовом документе мозаичное изображение как показано на рисунке



6. Напишите функцию, которая будет генерировать случайный пароль. В пароле должно быть от 8 до 15 символов Юникода из диапазонов 48-57 (цифры), 65-90 (буквы латинского алфавита в верхнем регистре) и 97-122 (буквы латинского алфавита в нижнем регистре). Сгенерируйте и выведите на экран три пароля.
7. Напишите функцию, которая будет строить круговую диаграмму. Постройте с помощью этой функции круговую диаграмму количества микроэлементов в болгарском перце
8. Напишите функцию для построения гистограммы, причем цвет столбцов



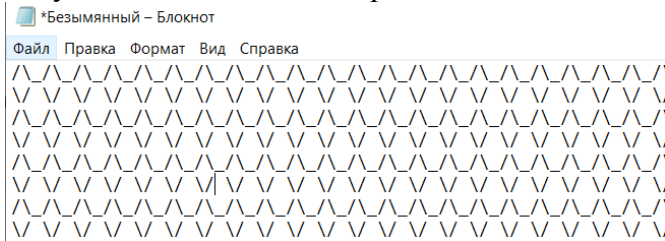
гистограммы должен градуироваться от максимального до минимального как . Напишите функцию, которая будет формировать список распределения частоты встречаемости различных букв в русскоязычном техническом тексте художественной литературы, объемом не менее 10000 символов. Постройте гистограмму. Сохраните как рисунок .jpg

Вариант 14

1. Напишите функцию, которая будет находить произведение всех элементов в списке
2. Напишите функцию, которая будет удалять из строки все согласные буквы, и записывать полученный результат в файл. Каждый запуск программы должен дозаписывать файл.
3. Напишите функцию для решения системы уравнений методом Крамера. Результат сохраняйте в файл. Вывести сообщение, если система уравнений не имеет решения. Система уравнений передается в функцию как три отдельных уравнения.

$$\begin{cases} x + 3y + 10z = 3 \\ 3x + 2y - 2z = 5 \\ 3x - 2y - 2z = 6 \end{cases}$$

4. Напишите функцию, которая будет строить графики в одном окне. Используйте линии разного цвета, постройте легенду. В качестве функций используйте уравнения из задания 3. z – параметр, который пользователь вводит с клавиатуры.
5. Напишите функцию, которая будет генерировать и сохранять в текстовом документе мозаичное изображение как показано на рисунке



6. Напишите функцию, которая будет генерировать случайный пароль. В пароле должно быть от 8 до 15 символов Юникода из диапазонов 48-57 (цифры), 65-90 (буквы латинского алфавита в верхнем регистре) и 97-122 (буквы латинского алфавита в нижнем регистре). Сгенерируйте и выведите на экран три пароля.
7. Напишите функцию, которая будет строить круговую диаграмму. Постройте с помощью этой функции круговую диаграмму содержания магния в различных продуктах
8. Напишите функцию для построения гистограммы, причем цвет столбцов



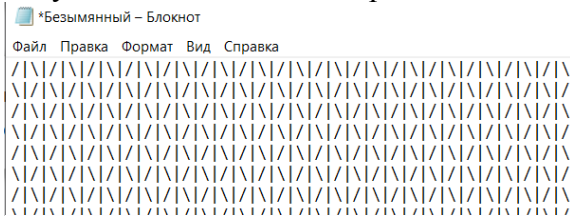
гистограммы должен градуироваться от максимального до минимального как . Напишите функцию, которая будет формировать список распределения частоты встречаемости различных букв в англоязычном техническом тексте, объемом не менее 10000 символов. Постройте гистограмму. Сохраните как рисунок .jpg

Вариант 15

1. Напишите функцию, которая будет суммировать все элементы в списке
2. Напишите функцию, которая будет заменять все гласные буквы на букву А, и записывать полученный результат в файл. Каждый запуск программы должен дозаписывать файл
3. Напишите функцию для решения системы уравнений методом Крамера. Результат сохраняйте в файл. Вывести сообщение, если система уравнений не имеет решения. Система уравнений передается в функцию как три отдельных уравнения.

$$\begin{cases} 4x + 4y - 8z = 2 \\ -2x - 4y + 3z = 4 \\ 3x - 2y - z = 6 \end{cases}$$

4. Напишите функцию, которая будет строить графики в одном окне. Используйте линии разного цвета, постройте легенду. В качестве функций используйте уравнения из задания 3. z – параметр, который пользователь вводит с клавиатуры.
5. Напишите функцию, которая будет генерировать и сохранять в текстовом документе мозаичное изображение как показано на рисунке



6. Напишите функцию, которая будет генерировать случайный пароль. В пароле должно быть от 8 до 15 символов Юникода из диапазонов 48-57 (цифры), 65-90 (буквы латинского алфавита в верхнем регистре) и 97-122 (буквы латинского алфавита в нижнем регистре). Сгенерируйте и выведите на экран три пароля.
7. Напишите функцию, которая будет строить круговую диаграмму. Постройте с помощью этой функции круговую диаграмму содержания калия в различных продуктах
8. Напишите функцию для построения гистограммы, причем цвет столбцов



гистограммы должен градуироваться от максимального до минимального как . Напишите функцию, которая будет формировать список распределения частоты встречаемости различных букв в русскоязычном тексте художественной литературы, объемом не менее 10000 символов. Постройте гистограмму. Сохраните как рисунок .jpg

Вариант 16

1. Напишите функцию, которая будет находить максимальный элемент в списке.
2. Напишите функцию, которая будет заменять все пробелы в строке на нижнее подчеркивание, и записывать полученный результат в файл. Каждый запуск программы должен дозаписывать файл
3. Напишите функцию для решения системы уравнений методом Крамера. Результат сохраняйте в файл. Вывести сообщение, если система уравнений не имеет решения. Система уравнений передается в функцию как три отдельных уравнения.

$$\begin{cases} 3x + 4y + 8z = 8 \\ x - 2y + 3z = 0 \\ 8x - 3y + 3z = 7 \end{cases}$$

- [illegible]

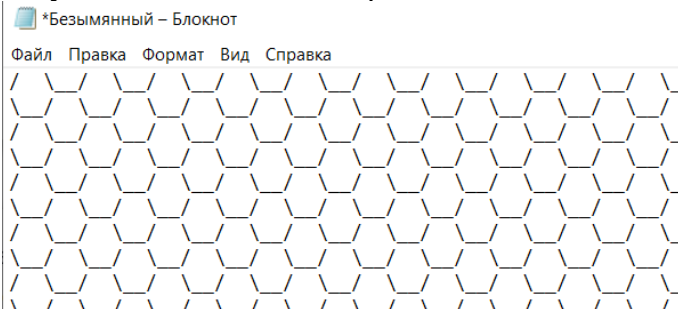
-

Вариант 17

1. Напишите функцию, которая будет находить минимальный элемент в списке
2. Напишите функцию, которая будет удалять все пробелы в строке, и записывать полученный результат в файл. Каждый запуск программы должен дозаписывать файл.
3. Напишите функцию для решения системы уравнений методом Крамера. Результат сохраняйте в файл. Вывести сообщение, если система уравнений не имеет решения. Система уравнений передается в функцию как три отдельных уравнения.

$$\begin{cases} x + y + 3z = 6 \\ 2x - 5y - z = 5 \\ x + 2y - 5z = -11 \end{cases}$$

4. Напишите функцию, которая будет строить графики в одном окне. Используйте линии разного цвета, постройте легенду. В качестве функций используйте уравнения из задания 3. z – параметр, который пользователь вводит с клавиатуры.
5. Напишите функцию, которая будет генерировать и сохранять в текстовом документе мозаичное изображение как показано на рисунке



6. Напишите функцию, которая будет генерировать случайный пароль. В пароле должно быть от 8 до 15 символов Юникода из диапазонов 48-57 (цифры), 65-90 (буквы латинского алфавита в верхнем регистре) и 97-122 (буквы латинского алфавита в нижнем регистре). Сгенерируйте и выведите на экран три пароля.
7. Напишите функцию, которая будет строить круговую диаграмму. Постройте с помощью этой функции круговую диаграмму вероятностей встречаемости гласных букв русского алфавита
8. Напишите функцию для построения гистограммы, причем цвет столбцов



гистограммы должен градуироваться от максимального до минимального как . Напишите функцию, которая будет формировать список распределения частоты встречаемости различных букв в англоязычном техническом тексте, объемом не менее 10000 символов. Постройте гистограмму. Сохраните как рисунок .jpg

Вариант 18

1. Напишите функцию, которая будет находить произведение всех элементов в списке
2. Напишите функцию, которая заменит в каждом слове первую букву на заглавную, и записывать полученный результат в файл. Каждый запуск программы должен дозаписывать файл
3. Напишите функцию для решения системы уравнений методом Крамера. Результат сохраняйте в файл. Вывести сообщение, если система уравнений не имеет решения. Система уравнений передается в функцию как три отдельных уравнения.

$$\begin{cases} 3x + 2y - z = 13 \\ 2x - y + 3z = -2 \\ x + 2y - z = 9 \end{cases}$$

4. Напишите функцию, которая будет строить графики в одном окне. Используйте линии разного цвета, постройте легенду. В качестве функций используйте уравнения из задания 3. z – параметр, который пользователь вводит с клавиатуры.
5. Напишите функцию, которая будет генерировать и сохранять в текстовом документе мозаичное изображение как показано на рисунке



6. Напишите функцию, которая будет генерировать случайный пароль. В пароле должно быть от 8 до 15 символов Юникода из диапазонов 48-57 (цифры), 65-90 (буквы латинского алфавита в верхнем регистре) и 97-122 (буквы латинского алфавита в нижнем регистре). Сгенерируйте и выведите на экран три пароля.
7. Напишите функцию, которая будет строить круговую диаграмму. Постройте с помощью этой функции круговую диаграмму численности населения в областных городах республики Беларусь.
8. Напишите функцию для построения гистограммы, причем цвет столбцов



гистограммы должен градуироваться от максимального до минимального как . Напишите функцию, которая будет формировать список распределения частоты встречаемости различных букв в англоязычном тексте художественной литературы, объемом не менее 10000 символов. Постройте гистограмму. Сохраните как рисунок .jpg

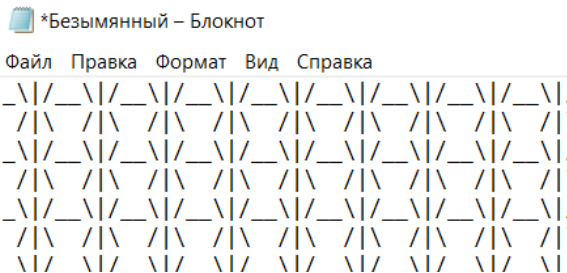
Вариант 19

1. Напишите функцию, которая будет суммировать все элементы в списке
2. Напишите функцию, которая будет добавлять к каждому слову строки порядковый номер, и записывать полученный результат в файл. Каждый запуск программы должен дозаписывать файл.

3. Напишите функцию для решения системы уравнений методом Крамера. Результат сохраняйте в файл. Вывести сообщение, если система уравнений не имеет решения. Система уравнений передается в функцию как три отдельных уравнения.

$$\begin{cases} x + 4y + 3z = 6 \\ 2x - 5y - z = 5 \\ x + 2y - 5z = -11 \end{cases}$$

4. Напишите функцию, которая будет строить графики в одном окне. Используйте линии разного цвета, постройте легенду. В качестве функций используйте уравнения из задания 3. z – параметр, который пользователь вводит с клавиатуры.
5. Напишите функцию, которая будет генерировать и сохранять в текстовом документе мозаичное изображение как показано на рисунке



6. Напишите функцию, которая будет генерировать случайный пароль. В пароле должно быть от 8 до 15 символов Юникода из диапазонов 48-57 (цифры), 65-90 (буквы латинского алфавита в верхнем регистре) и 97-122 (буквы латинского алфавита в нижнем регистре). Сгенерируйте и выведите на экран три пароля.
7. Напишите функцию, которая будет строить круговую диаграмму. Постройте с помощью этой функции круговую диаграмму химического состава клубники
8. Напишите функцию для построения гистограммы, причем цвет столбцов



гистограммы должен градуироваться от максимального до минимального как . Напишите функцию, которая будет формировать список распределения частоты встречаемости различных букв в русскоязычном тексте художественной литературы, объемом не менее 10000 символов. Постройте гистограмму. Сохраните как рисунок .jpg

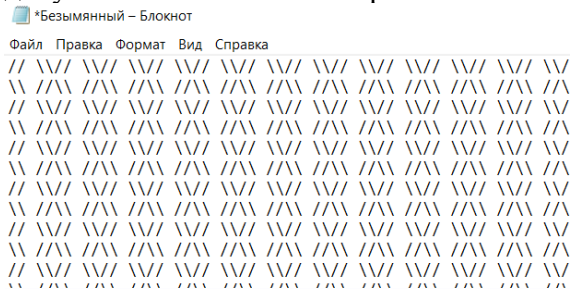
Вариант 20

1. Напишите функцию, которая будет находить максимальный элемент в списке
2. Напишите функцию, которая будет менять местами каждые два соседних символа, и записывать полученный результат в файл. Каждый запуск программы должен дозаписывать файл

3. Напишите функцию для решения системы уравнений методом Крамера. Результат сохраняйте в файл. Вывести сообщение, если система уравнений не имеет решения. Система уравнений передается в функцию как три отдельных уравнения.

$$\begin{cases} 3x + 2y - z = 13 \\ 2x - y + 3z = -2 \\ x + 2y - z = 9 \end{cases}$$

4. Напишите функцию, которая будет строить графики в одном окне. Используйте линии разного цвета, постройте легенду. В качестве функций используйте уравнения из задания 3. z – параметр, который пользователь вводит с клавиатуры.
5. Напишите функцию, которая будет генерировать и сохранять в текстовом документе мозаичное изображение как показано на рисунке



6. Напишите функцию, которая будет генерировать случайный пароль. В пароле должно быть от 8 до 15 символов Юникода из диапазонов 48-57 (цифры), 65-90 (буквы латинского алфавита в верхнем регистре) и 97-122 (буквы латинского алфавита в нижнем регистре). Сгенерируйте и выведите на экран три пароля.
7. Напишите функцию, которая будет строить круговую диаграмму. Постройте с помощью этой функции круговую диаграмму встречаемости золотоносных руд в 10 регионах Земли.
8. Напишите функцию для построения гистограммы, причем цвет столбцов



гистограммы должен градуироваться от максимального до минимального как . Напишите функцию, которая будет формировать список распределения частоты встречаемости различных букв в русскоязычном техническом тексте художественной литературы, объемом не менее 10000 символов. Постройте гистограмму. Сохраните как рисунок .jpg