

```
'''
```

1.2

Выведите на печать и определите тип переменной.

```
'''
```

```
x = 3+5.2*7
```

```
y = None
```

```
z = 'a', 5, 12.345, (2, 'b')
```

```
df = [['Антонова Антонина', 34, 'ж'], ['Борисов Борис', 26, 'м']]
```

```
A = {1, 'title', 2, 'content'}
```

```
print(f"{x} | {type(x)} \n{y} | {type(y)} \n{z} | {type(z)} \n{df} | {type(df)} \n{A} | {type(A)}")
```

**Program output:**

```
39.4 | <class 'float'>
```

```
None | <class 'NoneType'>
```

```
('a', 5, 12.345, (2, 'b')) | <class 'tuple'>
```

```
 [['Антонова Антонина', 34, 'ж'], ['Борисов Борис', 26, 'м']] | <class 'list'>
```

```
{1, 2, 'title', 'content'} | <class 'set'>
```

```
'''
```

1.3

Выведите на печать и определите тип переменной.

```
'''
```

```
x = 5 >= 2
```

```
A = {1, 3, 7, 8}
```

```
B = {2, 4, 5, 10, 'apple'}
```

```
C = A & B
```

```
df = 'Антонова Антолина', 34, 'Ж'
```

```
z = 'type'
```

```
D = [1, 'title', 2, 'content']
```

```
print(f"{x} | {type(x)} \n{A} | {type(A)} \n{B} | {type(B)} \n{C} | {type(C)} \n{df} | {type(df)} \n{z} | {type(z)} \n{D} | {type(D)}")
```

**Program output:**

```
True | <class 'bool'>
```

```
{8, 1, 3, 7} | <class 'set'>
```

```
{2, 4, 5, 10, 'apple'} | <class 'set'>
```

```
set() | <class 'set'>
```

```
('Антонова Антолина', 34, 'Ж') | <class 'tuple'>
```

```
type | <class 'str'>
```

```
[1, 'title', 2, 'content'] | <class 'list'>
```

```
'''
```

2.2

Вывести на экран является ли переменная x положительной, отрицательной или равна нулю.

```
'''
```

```
x = 125
```

```
if x<0:
```

```
    print("x отрицательный")
```

```
elif x==0:
```

```
    print("x равен нулю")
```

```
else:
```

```
    print("x положительный")
```

**Program output:**

x положительный

```
'''
```

2.3

Напишите код. задается  $x$ , напечатать какому из интервалов принадлежит:  $(-\infty, -5)$ ,  $[-5, 5]$  или от  $(5, +\infty)$

```
'''
```

```
x = 50
```

```
if x < -5:
```

```
    print("x принадлежит  $(-\infty, -5)$ ")
```

```
elif x <= 5:
```

```
    print("x принадлежит  $[-5, 5]$ ")
```

```
else:
```

```
    print("x принадлежит  $(5, +\infty)$ ")
```

**Program output:**

x принадлежит  $(5, +\infty)$

```
'''
```

3.2.1

Вывести Все числа от 1 до 10 с шагом 3.

```
'''
```

```
x = 1
```

```
while x<= 10:
```

```
    print(x)
```

```
    x+=3
```

**Program output:**

1

4

7

10

```
'''
```

### 3.2.2

Задан список. Напечатать Все элементы списка.

```
'''
```

```
models = ['KNN', 'decision tree', 'linear model']  
for model in models:  
    print(model)
```

**Program output:**

KNN

decision tree

linear model

```
'''
```

### 3.2.3

Создать список от 1 до 99 с шагом 7

```
'''
```

```
list_int = range(1, 100, 7)
```

```
print(list(list_int))
```

**Program output:**

```
[1, 8, 15, 22, 29, 36, 43, 50, 57, 64, 71, 78, 85, 92, 99]
```

```
'''
```

#### 3.2.4

Вывести на экран числа от 1 до 105 с шагом 25, используя функцию range()

```
'''
```

```
for i in range(5, 106, 25):  
    print(i)
```

**Program output:**

```
5  
30  
55  
80  
105
```



```
'''
```

### 3.2.5

Необходимо получить каждый k-й элемент массива a, начиная с элемента с индексом i (Включительно) и заканчивая элементом с индексом j (не Включительно), с шагом 2. Или переставить Все элементы В обратном порядке.

```
'''
```

```
a = [1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
b = a[1:7:2]
```

```
c = a[::-1]
```

```
print(b)
```

```
print(c)
```

**Program output:**

```
[2, 4, 6]
```

```
[9, 8, 7, 6, 5, 4, 3, 2, 1]
```

```
'''
```

### 3.3.1

Выведите числа из примера на while (3.2.1) в обратном порядке.

```
'''
```

```
x = 10
```

```
while x >= 1:
```

```
    print(x)
```

```
    x -= 3
```

**Program output:**

10

7

4

1

```
'''
```

### 3.3.2

При решении задач машинного обучения часто в качестве объектов исследования выступает человек. Создайте список значимых характеристик (признаков), идентифицирующих человека. Выведите список на экран.

```
'''
```

```
chara = ['злой', 'хаотичный', 'быстрый']  
for ch in chara:  
    print(ch)
```

**Program output:**

```
злой  
хаотичный  
быстрый
```

```
'''
```

### 3.3.3

Создать список чисел от 2 до 15 с шагом 1.

```
'''
```

```
list_int = range(2, 15, 1)
```

```
print(list(list_int))
```

**Program output:**

```
[2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14]
```

```
'''
```

3.3.4

Выведите числа из примера на for с функцией range() (3.2.4) в обратном порядке.

```
'''
```

```
for i in range(106, 5, -25):  
    print(i)
```

**Program output:**

106

81

56

31

6

```
'''
```

### 3.3.5

Напишите код, который Все элементы массива x с четными индексами переставит в обратном порядке. Т.е. если x = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9], то код должен сформировать [8, 1, 6, 3, 4, 5, 2, 7, 0, 9].

```
'''
```

```
a = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
b = a[::-2][::-1]
```

```
a[::-2] = b
```

```
print(b)
```

```
print(a)
```

**Program output:**

```
[8, 6, 4, 2, 0]
```

```
[8, 1, 6, 3, 4, 5, 2, 7, 0, 9]
```

```
'''
```

4.2.1

Рассчитать синус от  $e$  радиан и Вывести результат на экран

```
'''
```

```
import math as m
```

```
print (round(m.sin(m.e), 3))
```

**Program output:**

0.411

```
'''
```

#### 4.2.2

Для функции  $\sin(xe^{\cos(x)})$  на интервале  $[0;10]$  с шагом  $0,05$  построить график функции, закрасить площадь под ним и Вычислить значение этой площади.

Для Вычисления площади Воспользуйте функции `trapz()`, Вычисляющей интеграл по правилу трапеции. Для ее корректной работы необходимо подключить следующие библиотеки:

```
from scipy.integrate import simp
```

```
from numpy import trapz
```

```
'''
```

```
import numpy as np
```

```
import matplotlib.pyplot as plt
```

```
from scipy.integrate import simpson
```

```
from numpy import trapz
```

```
x = np.arange (0.0, 10, 0.1)
```

```
y = np.abs(np.sin(x*np.exp(np.cos(x))))
```

```
plt.grid()
```

```
plt.plot(x, y, c = "r")
```

```
plt.fill_between(x, y)
```

```
plt.savefig('4.2.2.png')
```

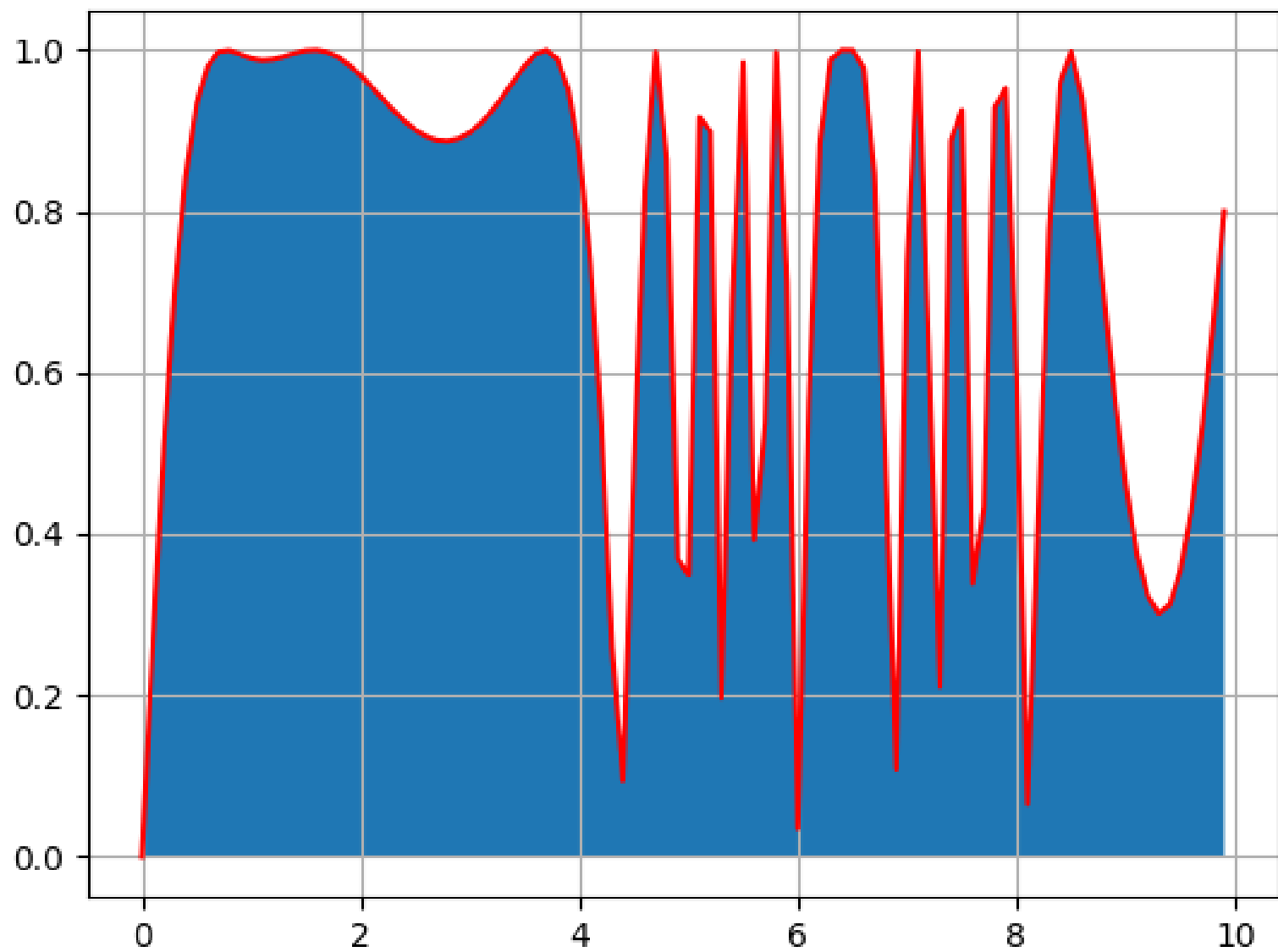
```
area = trapz(y,x)
```

```
print(area)
```

**Program output:**

7.350792907748665





```
'''
```

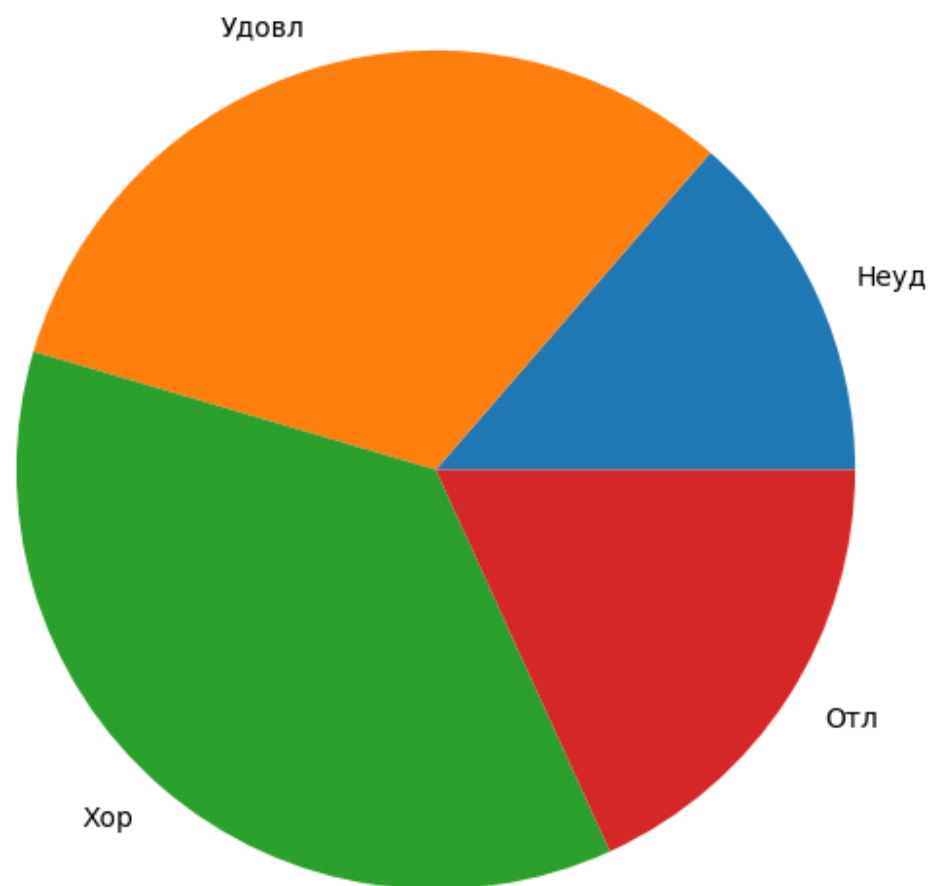
#### 4.2.3

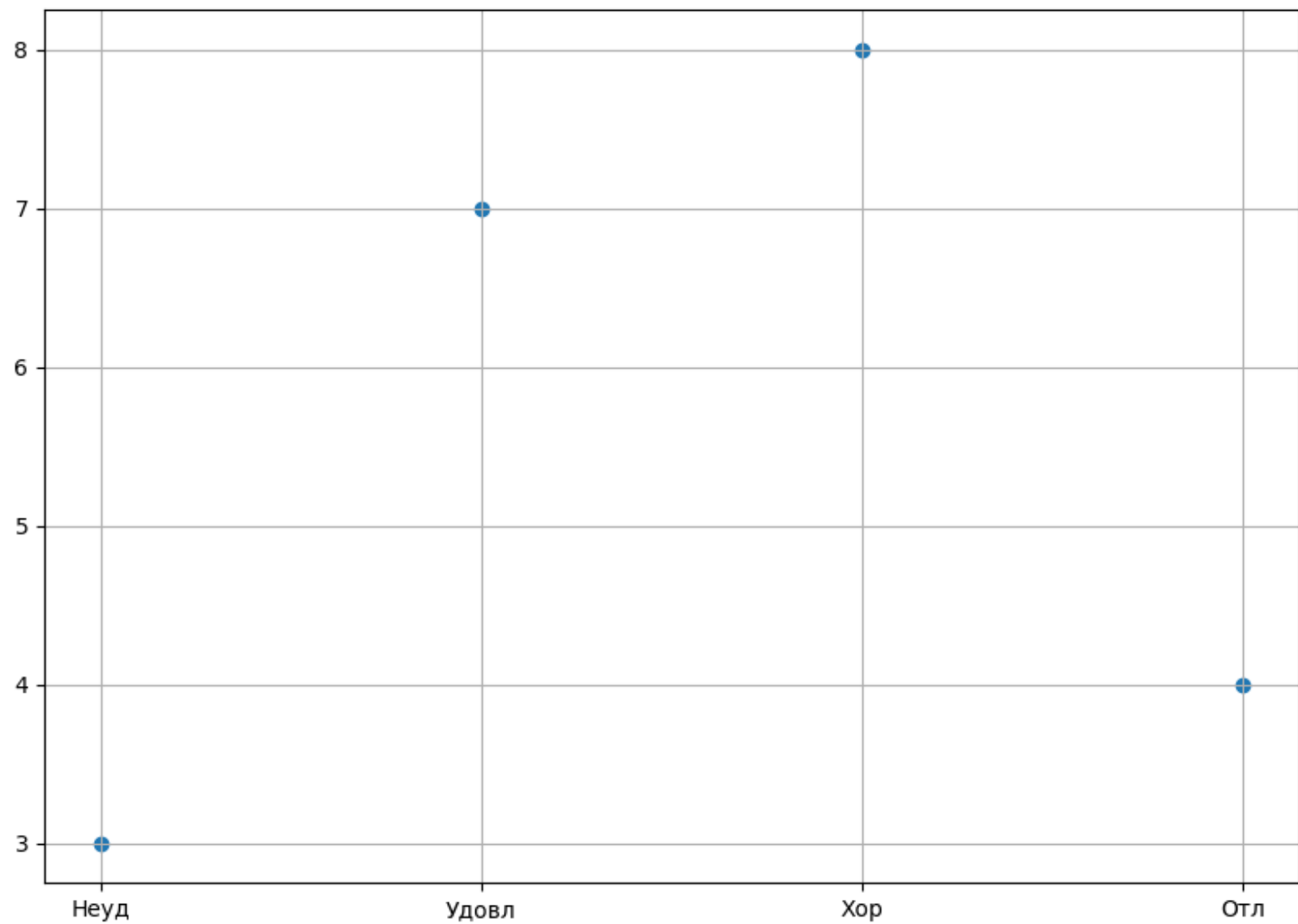
Дано некоторое распределение оценок в группе за экзамен. Нарисовать круговую и точечную диаграммы распределения.

```
'''
```

```
from matplotlib import pyplot as plt
import numpy as np
marks = ['Неуд', 'Удовл', 'Хор', 'Отл']
data = [3, 7, 8, 4]
fig = plt.figure(figsize = (10, 7))
plt.pie(data, labels = marks)
plt.savefig('4.2.3.1.png')
plt.clf()
plt.grid()
plt.scatter(marks, data)
plt.savefig('4.2.3.2.png')
```

**Program output:**





```
'''
```

#### 4.3.1

Задайте массив случайных значений из интервала(0; 1). Рассчитайте средние и медианные значения для массива, сравните результаты, какие Выводы можно сделать о значениях? Постройте точечную диаграмму рассеяния полученного ряда.

```
'''
```

```
import random
import numpy as np
import matplotlib.pyplot as plt
x = []
sum = 0
for i in range(1,10):
    x.append(round(random.random(),2))
    sum += x[i-1]

print(x)
print(f"Ср.знач.: {round(sum/len(x),2)}")
print(f"Meg.знач.: {round(np.median(x),2)}")

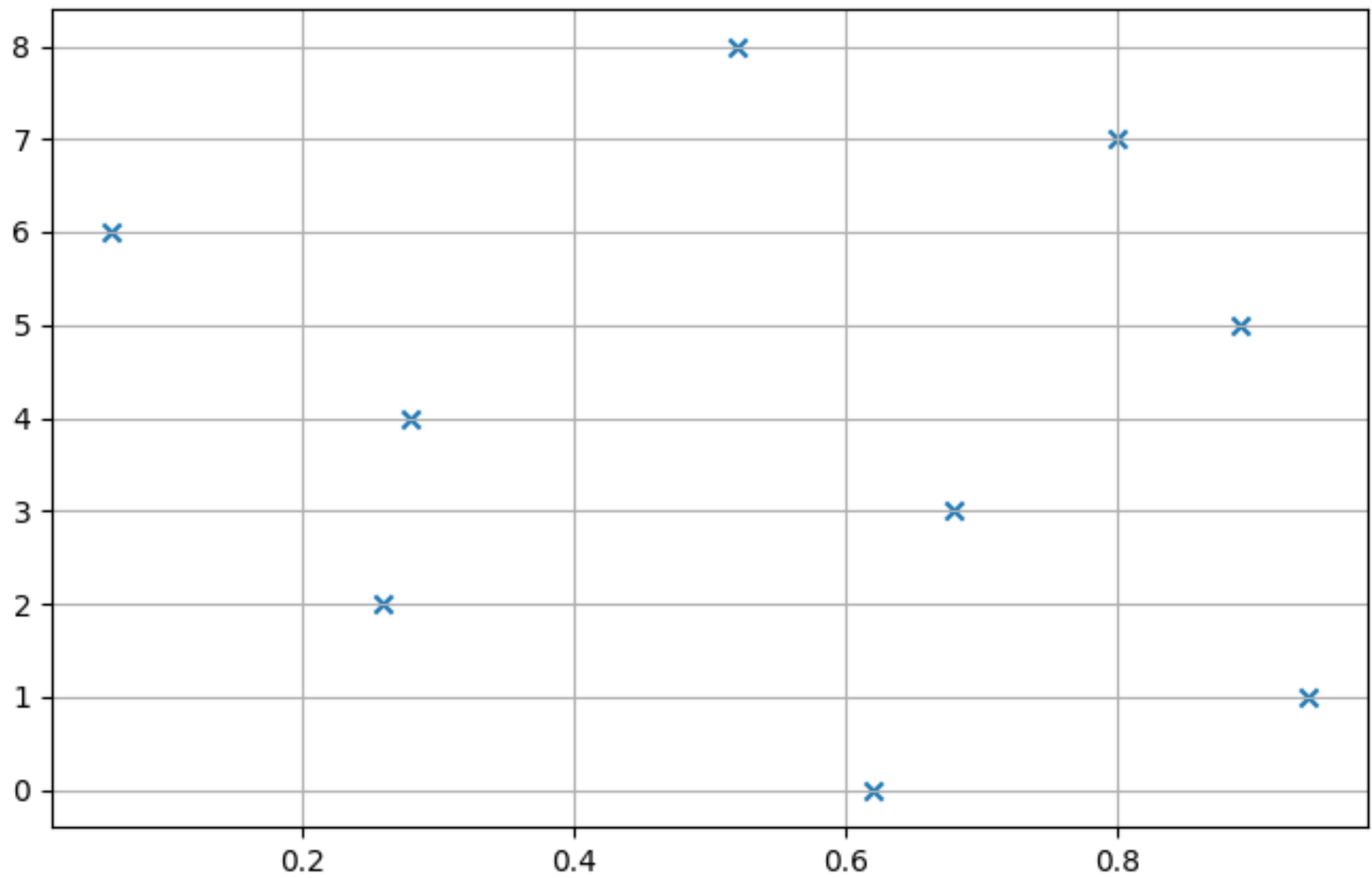
y = [y for y in range(9)]
plt.figure(figsize=(8,5))
plt.scatter(x, y, marker = "x")
plt.grid(True)
plt.savefig("4.3.1.png")
```

#### Program output:

```
[0.62, 0.94, 0.26, 0.68, 0.28, 0.89, 0.06, 0.8, 0.52]
```

```
Ср.знач.: 0.56
```

```
Meg.знач.: 0.62
```



```
'''
```

#### 4.3.2

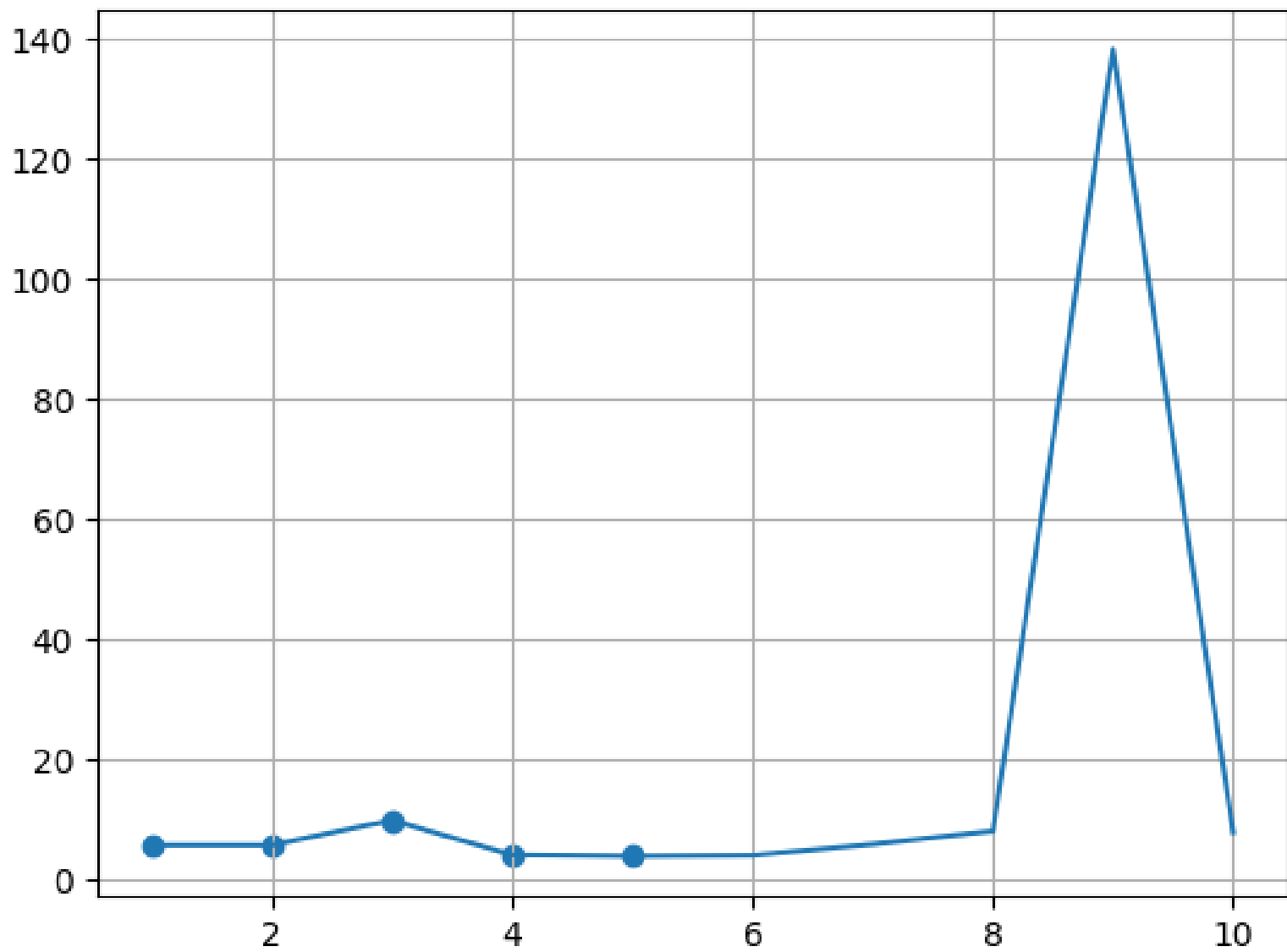
Создать массив из 10 значений функции (x, например, изменяется от 1 до 10). Выделить срез первой половины массива и построить графики для основного массива – линейный и для среза – точечный

```
'''
```

```
import random
import numpy as np
import matplotlib.pyplot as plt
x=[]
y=[]
for i in range(10):
    x.append(i+1)
    y.append((np.sqrt(1+np.exp(np.sqrt(x[i-1])))+np.cos(x[i-1]*x[i-1]))/np.abs(1-(np.sin(x[i-1])**3)))+np.log(abs(2*x[i-1])))

b=x[:5]
plt.plot(x,y)
plt.scatter(b,y[:5])
plt.grid(True)
plt.savefig("4.3.2.png")
```

**Program output:**





```
...
```

#### 4.3.3

Построить график на интервале  $(0,10)$  с шагом 1 с заливкой площади и найти эту площадь пог ним. Для Вычисления площади Воспользуйте функции `trapz()`, Вычисляющей интеграл по правилу трапеции.

Для ее корректной работы необходимо подключить следующие библиотеки:

```
from scipy.integrate import simp
```

```
from numpy import trapz
```

```
...
```

```
import numpy as np
```

```
import matplotlib.pyplot as plt
```

```
from scipy.integrate import simpson
```

```
from numpy import trapz
```

```
x = np.arange (0.0, 10, 1)
```

```
y = np.abs(np.cos(x*np.exp(np.cos(x)+np.log(x+1))))
```

```
plt.grid()
```

```
plt.plot(x, y, c = "r")
```

```
plt.fill_between(x, y)
```

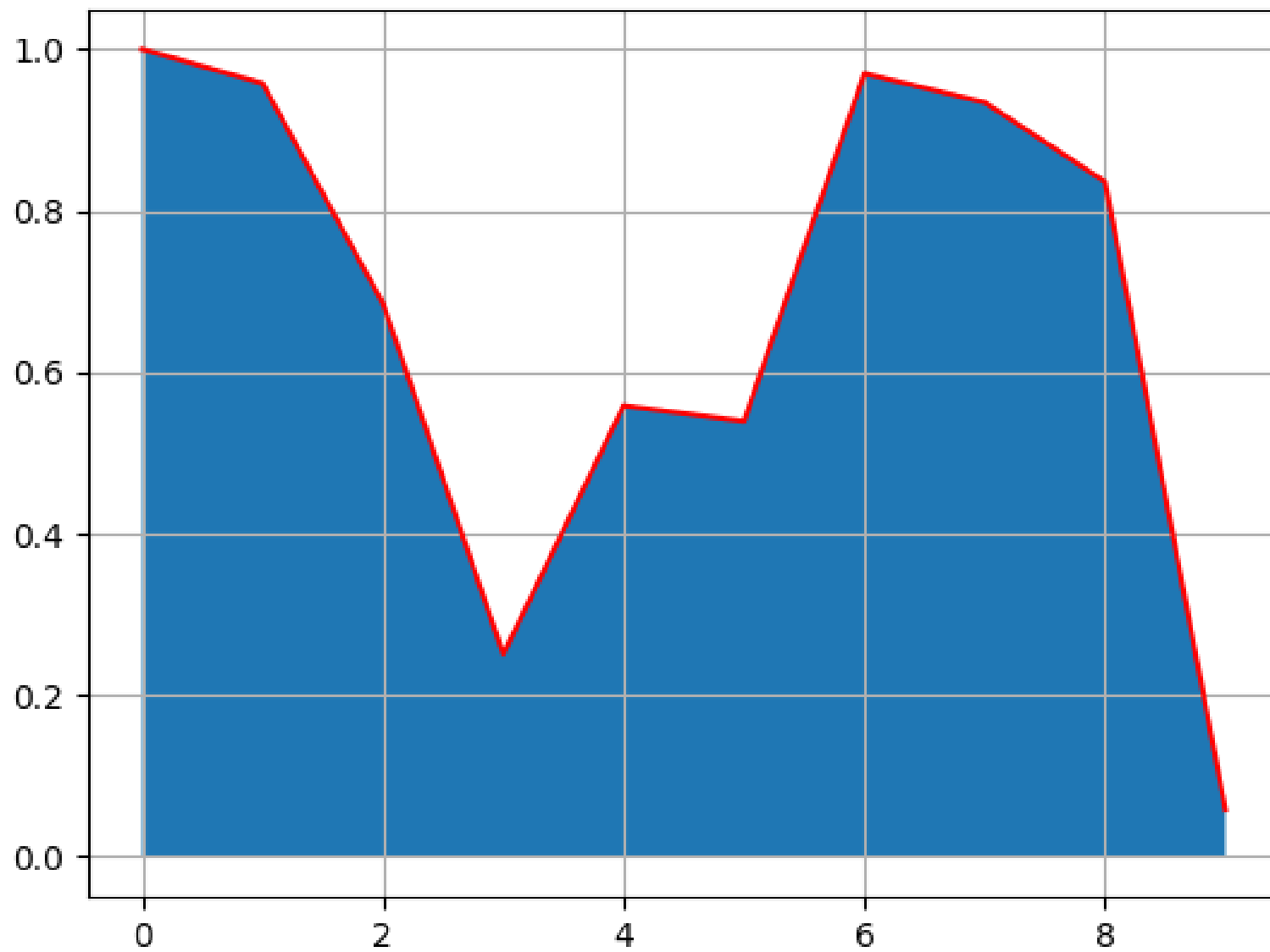
```
plt.savefig('4.3.3.png')
```

```
area = trapz(y,x)
```

```
print(area)
```

**Program output:**

6.26034587549474



```
'''
```

#### 4.3.4

Откройте статистические данные по стоимости акций Apple, Microsoft, Google за 12 месяцев 2021 г. Постройте 3 графика на плоскости и оцените их динамику. Подумайте, как можно улучшить читаемость результатов.

Для начальных данных допустимо использовать следующий ресурс:

<https://finance.yahoo.com>

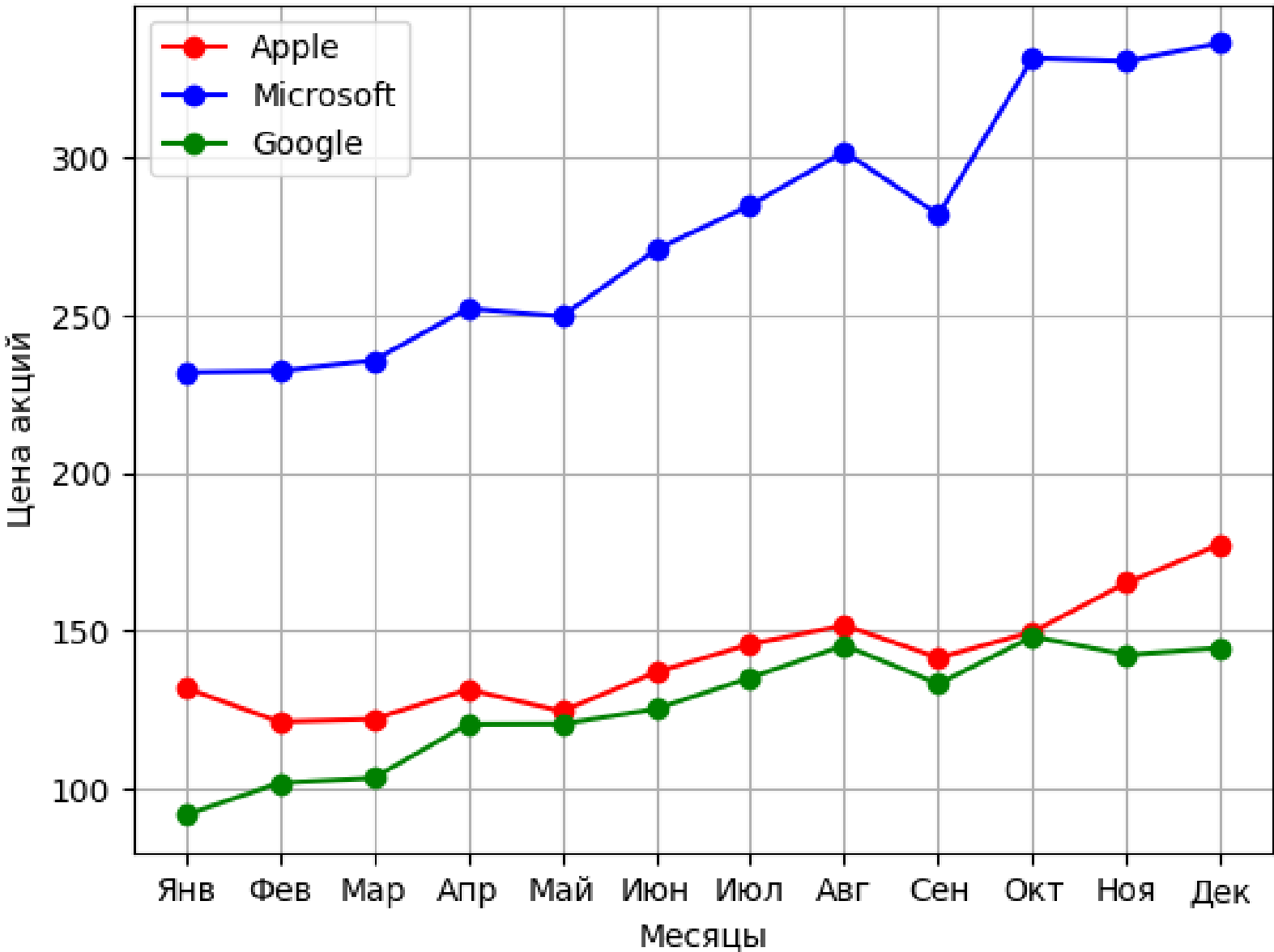
```
'''
```

```
import math as m
import matplotlib.pyplot as plt

apple = [131.96, 121.26, 122.15, 131.46, 124.61, 136.96, 145.86, 151.83, 141.50, 149.80, 165.30, 177.57]
microsoft = [231.96, 232.38, 235.77, 252.18, 249.68, 270.90, 284.91, 301.88, 281.92, 331.62, 330.59, 336.32]
google = [91.79, 101.84, 103.43, 120.51, 120.58, 125.32, 135.22, 145.46, 133.27, 148.27, 142.45, 144.68]
x=[i for i in range(1,13)]
plt.grid()
plt.plot(x, apple, label='Apple', color='r', marker='o')
plt.plot(x, microsoft, label='Microsoft', color='b', marker='o')
plt.plot(x, google, label='Google', color='g', marker='o')
plt.title('Динамика цен акций Apple, Microsoft и Google за 12 месяцев в 2021 году')
plt.xlabel('Месяцы')
plt.ylabel('Цена акций')
plt.xticks(x, ['Янв', 'Фев', 'Мар', 'Апр', 'Май', 'Июн', 'Июл', 'Авг', 'Сен', 'Окт', 'Ноя', 'Дек'])
plt.legend()
plt.savefig('4.3.4.png')
```

Program output:

Динамика цен акций Apple, Microsoft и Google за 12 месяцев в 2021 год



```
'''
```

#### 4.3.5

Создайте простейший калькулятор, включающий основные действия для двух переменных '+', '-', '\*', '/', а также Вычисление следующих функций:  $e^{x+y}$ ,  $\sin(x+y)$ ,  $\cos(x+y)$ ,  $xy$ .

```
'''
```

```
import numpy as np
x = float(input('x: '))
y = float(input('y: '))
a = int(input("Choose function:\n1: +\n2: -\n3: *\n4: /\n5: e(x+y)\n6: sin(x+y)\n7: cos(x+y)\n8: x^y\n"))

print("Result:")
match a:
    case 1:
        print(x+y)
    case 2:
        print(x-y)
    case 3:
        print(x*y)
    case 4:
        if y == 0:
            print("Dividing by zero is prohibited")
        else:
            print(x/y)
    case 5:
        print(np.exp(x+y))
    case 6:
        print(np.sin(x+y))
    case 7:
        print(np.cos(x+y))
    case 8:
        print(x**y)
```

#### Program output:

x: y: Choose function:

1: +

2: -

3: \*

4: /

5:  $e^{(x+y)}$

6:  $\sin(x+y)$

7:  $\cos(x+y)$

8:  $x^y$

Result:

Dividing by zero is prohibited