**Министерство науки и высшего образования РФ**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский Авиационный Институт»**

**Национальный Исследовательский Университет**

**Институт** №8 «Информационные технологии и прикладная математика»

**Кафедра** 806 «Вычислительная математика и программирование»

**Лабораторная работа №1**

**по курсу «Компьютерная графика»**

|  |  |
| --- | --- |
| Студент: | Попов И. П. |
| Группа: | М8О-306Б-20 |
| Преподаватель: | Филиппов Г. С. |
| Подпись: |  |
| Оценка: |  |
| Дата: |  |

# Лабораторная работа №1

**Тема:** Построение изображений 2D- кривых.

**Задание:** Написать и отладить программу, строящую изображение заданной замечательной кривой. Обеспечить автоматическое масштабирование и центрирование кривой при изменении размеров окна.

**Вариант: ¸**

x, y – декартовы координаты

a, A, B, - константы, значения которых выбираются пользователем (вводятся в окне программы) a > 0

# Описание

Программа написана на языке программирования Python с использованием библиотеки mathplotlib для отрисовки двумерного графика.

Для того, чтобы задать значение параметра a, используется специальный текстовое поле: TextBox из mathplotlib.widgets. При изменении пользователем параметра в этом поле, пересчитываются значения координат, по ним перестраивается график, таким образом происходит обновление изображения.

# Исходный код:

'''

Popov Ilya

M80-306Б-20

y^2 = x^2/((a-x)(a+x))

Тема: Построение изображений 2D- кривых.

Задание: Написать и отладить программу, строящую изображение заданной замечательной кривой.

Обеспечить автоматическое масштабирование и центрирование кривой при изменении размеров окна.

'''

import numpy as np

import matplotlib.pyplot as plt

from matplotlib.widgets import TextBox

a = 1

A = -0.5

B = 0.5

x = np.linspace(A, B, int((B-A) \* 100))

y = x \* np.sqrt(1 / ((a-x)\*(a+x)))

color = 'blue'

style = '-'

def update\_function():

    x = np.linspace(A, B, max(0,  int((B-A) \* 100)))

    y = x \* np.sqrt(1 / ((a-x)\*(a+x)))

    l.set\_xdata(x)

    l.set\_ydata(y)

    ax.set\_xlim(

        np.min(x) - (np.max(x) - np.min(x)) / 100,

        np.max(x) + (np.max(x) - np.min(x)) / 100

    )

    ax.set\_ylim(

        np.min(y) - (np.max(y) - np.min(y)) / 100,

        np.max(y) + (np.max(y) - np.min(y)) / 100

    )

    plt.setp(l, color = color, linestyle = style) #свойства для графика

fig = plt.figure()

#Положение нижнего края вложенных графиков в процентах от высоты рисунка

fig.subplots\_adjust(bottom=0.2)

# Удалить все привязки клавиш по умолчанию

fig.canvas.mpl\_disconnect(fig.canvas.manager.key\_press\_handler\_id)

#параметры subplot

ax = fig.add\_subplot(111)

ax.set\_title(r"$y^2 = x^2/((a-x)(a+x))$")

l, = ax.plot(x, y)

ax.grid() #сетка

ax.set\_xlim(

    np.min(x) - (np.max(x) - np.min(x)) / 100,

    np.max(x) + (np.max(x) - np.min(x)) / 100

)

ax.set\_ylim(

    np.min(y) - (np.max(y) - np.min(y)) / 100,

    np.max(y) + (np.max(y) - np.min(y)) / 100

)

def submit\_a(a\_):

    global a

    a = float(a\_)

    update\_function()

    plt.draw()

axbox\_a = fig.add\_axes([0.1, 0.05, 0.1, 0.06]) #[left, bottom, width, height] относительный размер

text\_box\_a = TextBox(axbox\_a, "a ")

text\_box\_a.on\_submit(submit\_a)

text\_box\_a.set\_val(a)

def submit\_A(A\_):

    global A

    A = float(A\_)

    update\_function()

    plt.draw()

axbox\_A = fig.add\_axes([0.25, 0.05, 0.1, 0.06])

text\_box\_A = TextBox(axbox\_A, "A ")

text\_box\_A.on\_submit(submit\_A)

text\_box\_A.set\_val(A)

def submit\_B(B\_):

    global B

    B = float(B\_)

    update\_function()

    plt.draw()

axbox\_B = fig.add\_axes([0.4, 0.05, 0.1, 0.06])

text\_box\_B = TextBox(axbox\_B, "B ")

text\_box\_B.on\_submit(submit\_B)

text\_box\_B.set\_val(B)

def submit\_color(color\_):

    global color

    color = str(color\_)

    update\_function()

    plt.draw()

axbox\_C = fig.add\_axes([0.6, 0.05, 0.1, 0.06])

text\_box\_C = TextBox(axbox\_C, "Color: ")

text\_box\_C.on\_submit(submit\_color)

text\_box\_C.set\_val(color)

def submit\_style(style\_):

    global style

    style = str(style\_)

    update\_function()

    plt.draw()

axbox\_S = fig.add\_axes([0.8, 0.05, 0.1, 0.06])

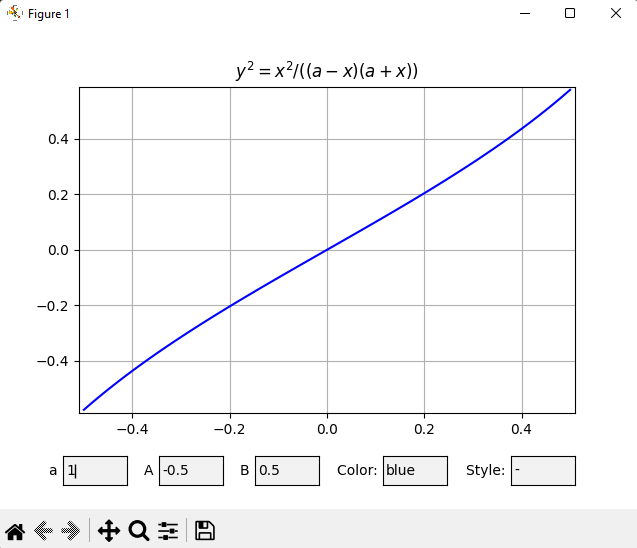
text\_box\_S = TextBox(axbox\_S, "Style: ")

text\_box\_S.on\_submit(submit\_style)

text\_box\_S.set\_val(style)

plt.show()

# Работа программы:

****

# Выводы:

В ходе выполнения данной лабораторной работы была написана программа на языке Python для получения графика функции ****, а также функция ее перестройки при изменении пользователем константы а и рассматриваемого интервала (A; B).

Библиотека MatPlotLib – очень удобный инструмент для решения задач на построение 2D – графиков.