

**Московский авиационный институт
(Национальный исследовательский университет)**

Факультет: «Информационные технологии и прикладная математика»

Кафедра: 806 «Вычислительная математика и программирование»

Дисциплина: «Компьютерная графика»

Лабораторная работа № 1

Тема: Построение изображений 2D- кривых.

Студентка: Довженко А.А.

Группа: 80-307

Преподаватель: Чернышов Л.Н.

Дата:

Оценка:

Москва, 2018

1. Постановка задачи

Написать и отладить программу, строящую изображение заданной замечательной кривой.

Вариант 7.

$$x = a \cdot \cos(t)$$

$$y = b \cdot \sin(t)$$

x , y – декартовы координаты; t – независимый параметр; a , b – константы, значения которых выбираются пользователем (вводятся в окне программы). a , $b > 0$.

Обеспечить автоматическое масштабирование и центрирование кривой при изменении размеров окна.

2. Решения задачи

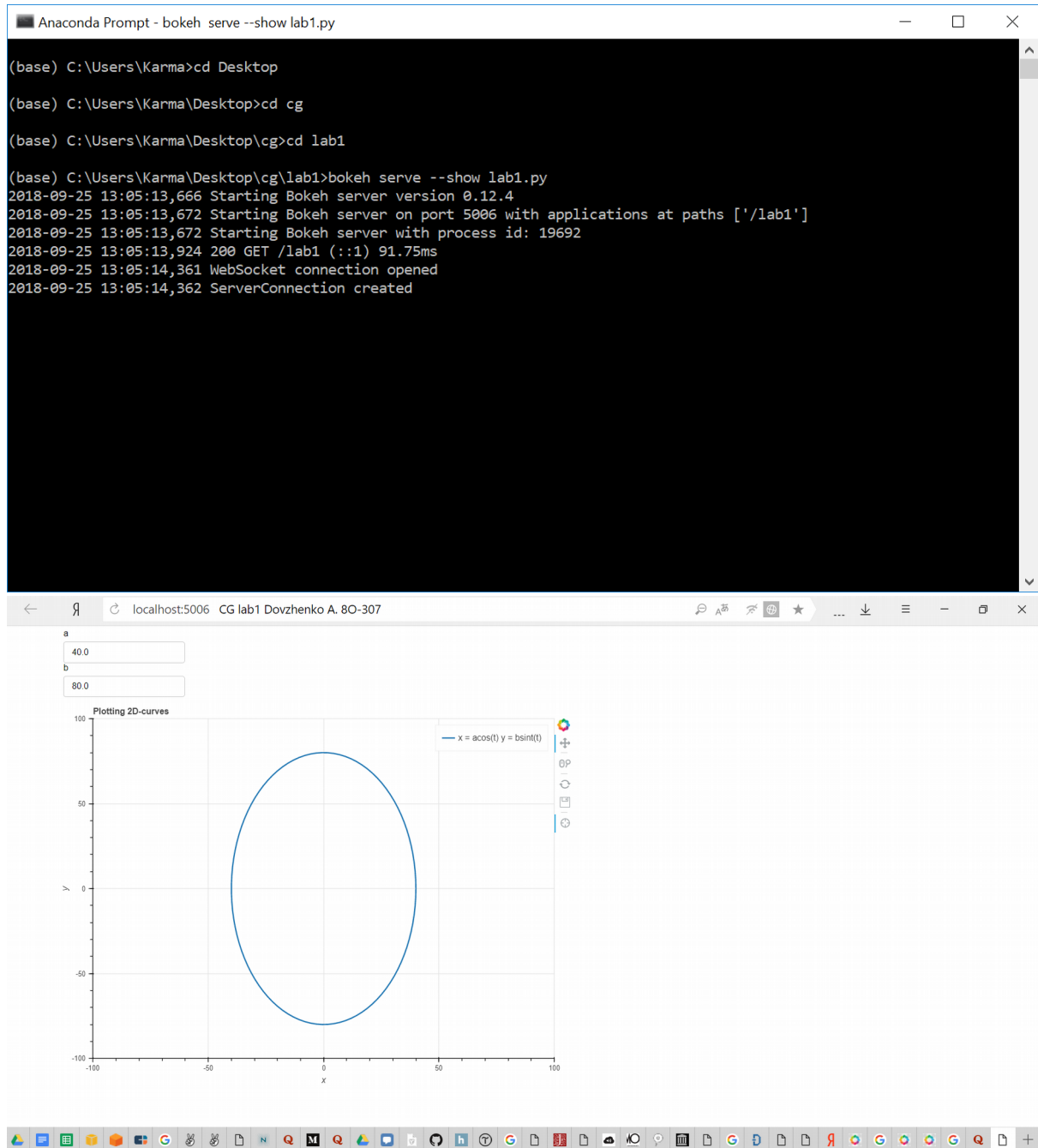
Для решения задачи я проанализировала возможные варианты реализации на чистом python. Мной были обнаружены следующие пути имплементации:

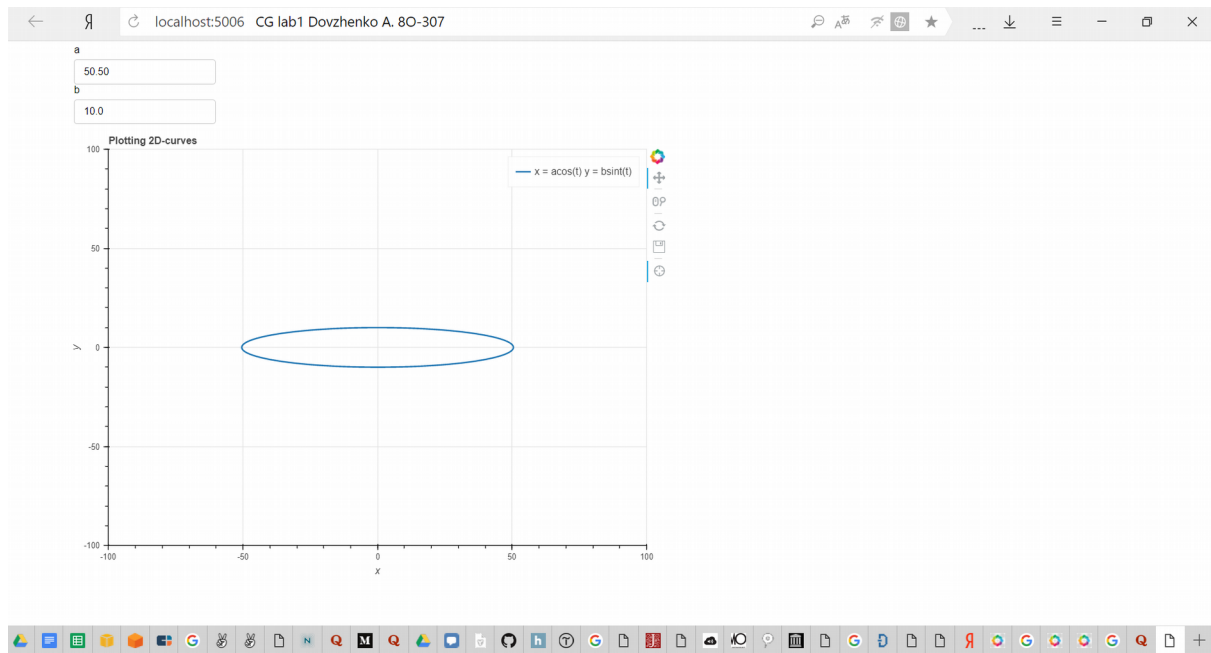
- 1) использовать модули matplotlib и mpld3. Первый – для создания кривой, второй – для отображения её в браузере. В этом случае мне пришлось бы писать отдельный плагин на js для обновления значений, вводимых пользователей. Мне этот вариант не подошёл, потому что мне хотелось написать приложение исключительно на python.
- 2) написать клиент-серверное приложение, клиент на js, бэк на python, связать с помощью flask. Эта идея была отклонена по тем же причинам, что и предыдущая.
- 3) использовать jupyter для отображения кода на python в браузере. Но, подумав о масштабировании, я пришла к выводу, что сложно будет развернуть тетради на AWS.
- 4) использовать bokeh. Я не нашла минусов у этого модуля на данном этапе изучения компьютерной графики, поэтому использовала его.

Я создала координатную плоскость и строю график, предлагаемый по умолчанию, со значениями констант 1.0. Далее построила два поля для ввода, куда пользователь может записать свои значения констант. Они обновляются благодаря функции `update_data`, которая перестраивает график при каждом обновлении констант.

3. Руководство по использованию программы

pip install bokeh
bokeh serve --show <filename.py>





4. Листинг программы

'''

Довженко А.А. М8О-307Б

Лабораторная работа №1.

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

Вариант №7:

$$x = a * \cos(t)$$

$$y = b * \sin(t)$$

'''

```
from bokeh.io import curdoc
from bokeh.layouts import row, column, widgetbox
from bokeh.models import ColumnDataSource
from bokeh.models.widgets import TextInput
from bokeh.plotting import Figure
import numpy as np
from math import pi
```

```
def update_data(attr, old, new):
```

```
    try:
```

```
        x = float(a.value) * np.cos(t)
```

```
        y = float(b.value) * np.sin(t)
```

```
        source.data = dict(x=x, y=y)
```

```

except ValueError:
    pass

t = np.linspace(0, 2*pi, 100)
x = np.cos(t)
y = np.sin(t)

source = ColumnDataSource(data=dict(x=x, y=y))

plot = Figure(plot_width=800, x_range=(-100, 100), y_range=(-100, 100),
tools="crosshair,pan,reset,save,wheel_zoom")
plot.xaxis.axis_label="x"
plot.yaxis.axis_label="y"
plot.line(x="x", y="y", source=source, line_width=2, legend="x = cos(t)\ny = sin(t)")
plot.title.text = "Plotting 2D-curves"

a = TextInput(title="a", value='1.0')
b = TextInput(title="b", value='1.0')

for w in [a, b]:
    w.on_change('value', update_data)

inputs = widgetbox(a, b)
curdoc().add_root(row(inputs, plot, width=800))
curdoc().title = "CG lab1 Dovzhenko A. 8O-307"

```

5. Выводы

Я познакомилась с модулем `bokeh`, который позволяет строить разные виды графиков. С помощью `bokeh` можно построить как статические графики, так и интерактивные, который и был реализован в лабораторной. Знания, полученные в ходе выполнения работы, несомненно понадобятся мне, потому что задачи визуализации данных встречаются довольно часто. К сожалению, развернуть приложение на AWS мне не удалось, но я обязательно сделаю это в следующих лабораторных работах.

6. Используемые источники

1. Документация модуля bokeh [Электронный ресурс]. URL: <https://bokeh.pydata.org/en/latest/> (дата обращения: 16.09.2018).
2. Исходный код модуля bokeh [Электронный ресурс]. URL: <https://github.com/bokeh> (дата обращения: 20.09.2018).