

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

Институт: «Информационные технологии и прикладная математика»

Кафедра: 806 «Вычислительная математика и программирование»

Дисциплина: «Компьютерная графика»

Лабораторная работа № 7

**Тема: Построение плоских полиномиальных
кривых.**

Студент: Махмудов Орхан

Группа: О8-305

Преподаватель: Чернышов Л.Н.

Дата:

Оценка:

Москва, 2020

1. Постановка задачи

Тема: Тема: Построение плоских полиномиальных кривых.

Задание: Написать программу, строящую полиномиальную кривую по заданным точкам. Обеспечить возможность изменения позиции точек и, при необходимости, значений касательных векторов и натяжения.

Вариант №6: Кривая Безье 4-й степени

Решение задачи:

Кривая Безье — параметрическая [кривая](#), задаваемая выражением

$$\mathbf{B}(t) = \sum_{k=0}^n \mathbf{P}_k \mathbf{b}_{k,n}(t), \quad 0 \leq t \leq 1$$

где \mathbf{P}_k — контрольные точки, определяющие форму сплайна Безье, а $\mathbf{b}_{k,n}(t)$ — базисные функции кривой Безье, называемые также [полиномами Бернштейна](#).

$$\mathbf{b}_{k,n}(t) = \binom{n}{k} t^k (1-t)^{n-k},$$

где $\binom{n}{k} = \frac{n!}{k!(n-k)!}$ — число [сочетаний](#) из n по k , где n — степень полинома, k — порядковый номер опорной вершины.

В нашем случае $n = 4$, так как кривая Безье 4-ой степени

Формулы взяты из этой статьи [Кривая Безье](#) .

2. Описание программы

Используемая среда: Visual Studio Code

Используемые библиотеки: Matplotlib, Numpy, Tkinter

Язык программирования: Python

Используемые структуры данных: массивы

Ввод: Все координаты точек кривой вводятся через форму пользователем

Вывод: Выводится кривая по заданным точкам

Краткая инструкция для пользователя:

Запуск программы по кнопке в среде разработки, после запуска появляется

форма, в которой нужно вбить координаты точек. После того, как координаты введены, нужно нажать на кнопку “Запуск”, после нажатия появится окно с графиком.

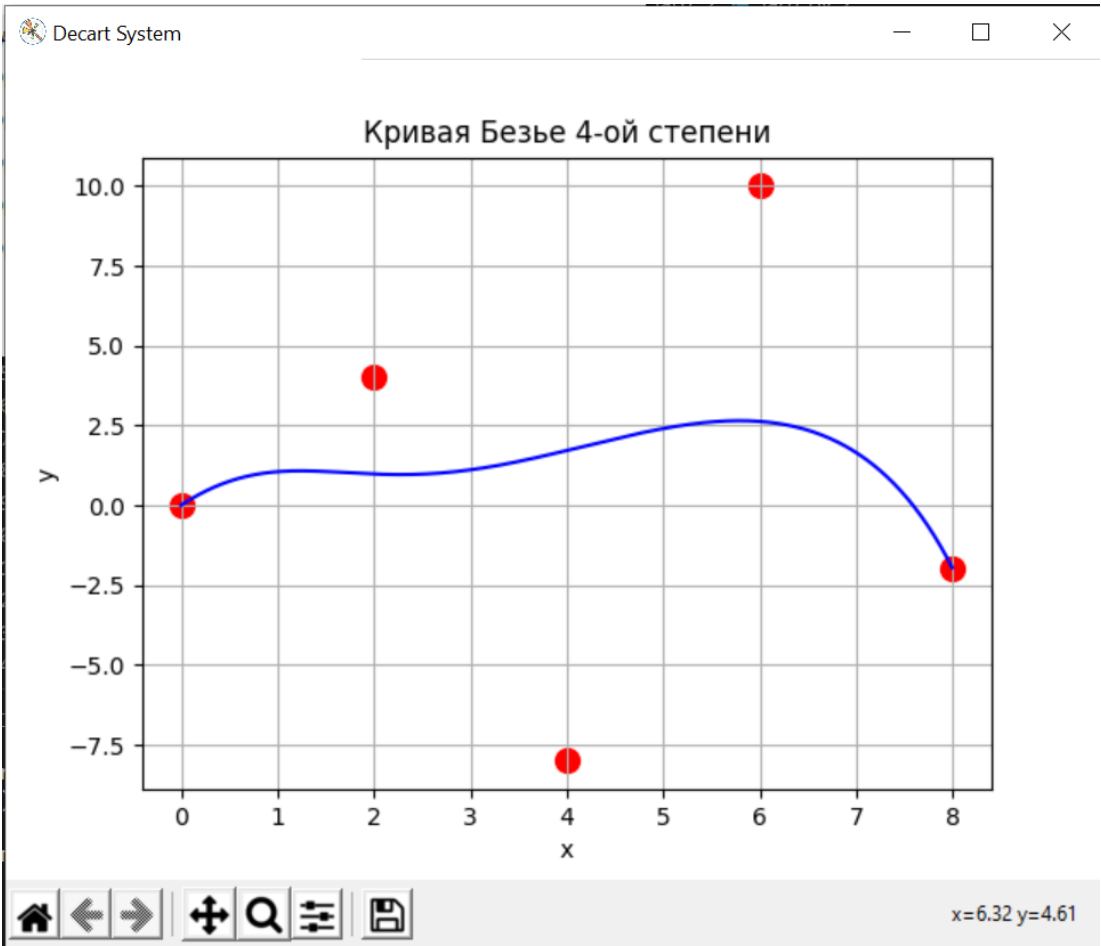
3. Набор тестов

№ теста	Координата первой точки	Координата второй точки	Координата третьей точки	Координата четвёртой точки	Координата пятой точки
1	(0,0)	(2,4)	(4,-8)	(6,10)	(8,-2)
2	(0,0)	(1,40)	(3.5,25)	(3,-10)	(4,2)
3	(0,0)	(0,4)	(5, -10)	(10,4)	(10,0)

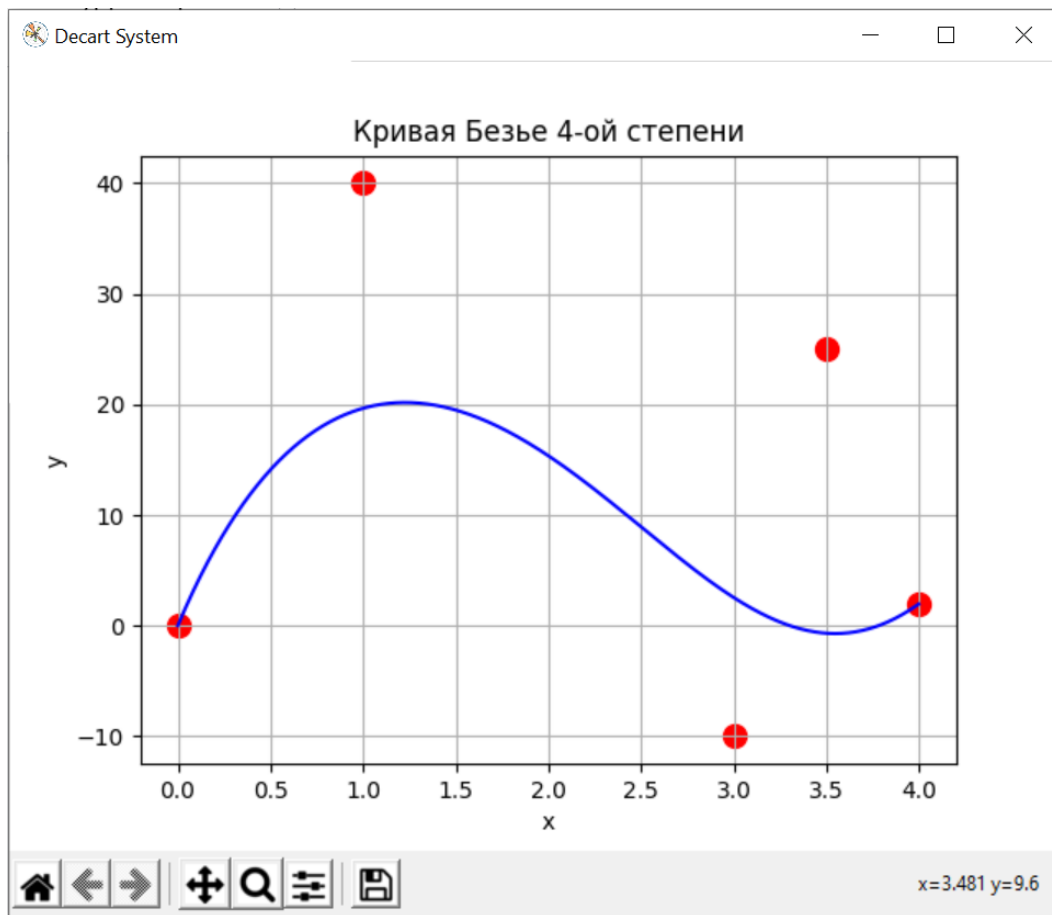
4. Результаты выполнения тестов

№ теста	Координата первой точки	Координата второй точки	Координата третьей точки	Координата четвёртой точки	Координата пятой точки
---------	-------------------------	-------------------------	--------------------------	----------------------------	------------------------

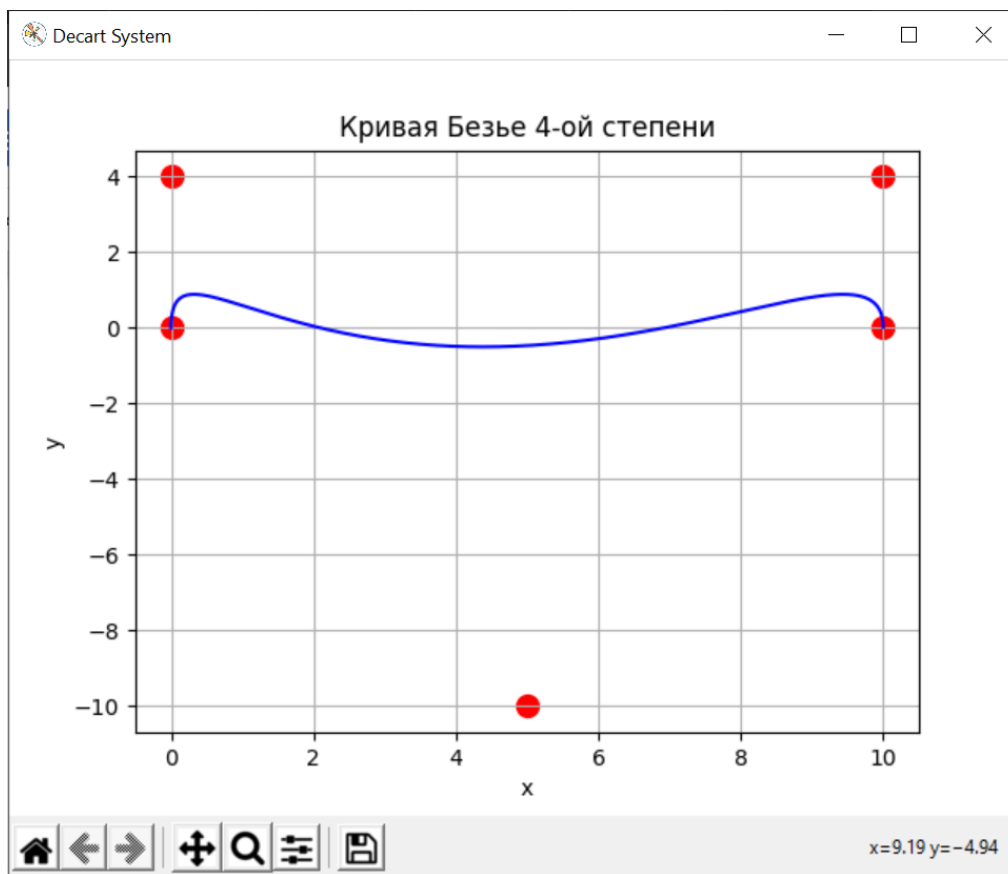
1	(0,0)	(2,4)	(4,-8)	(6,10)	(8,-2)
---	-------	-------	--------	--------	--------



№ теста	Координата первой точки	Координата второй точки	Координата третьей точки	Координата четвёртой точки	Координата пятой точки
2	(0,0)	(1,40)	(3.5,25)	(3,-10)	(4,2)



№ теста	Координата первой точки	Координата второй точки	Координата третьей точки	Координата четвёртой точки	Координата пятой точки
3	(0,0)	(0,4)	(5, -10)	(10,4)	(10,0)



5. Листинг программы

```
#Махмудов Орхан группа М8О-305В-18
#Кривая Безье 4-ой степени
import numpy as np
import matplotlib.pyplot as plt
from tkinter import Tk, Label, Button, Entry
def clicked():
    x1 = float(txt1.get())
    y1 = float(txt2.get())
    x2 = float(txt3.get())
    y2 = float(txt4.get())
    x3 = float(txt5.get())
    y3 = float(txt6.get())
    x4 = float(txt7.get())
    y4 = float(txt8.get())
    x5 = float(txt9.get())
    y5 = float(txt10.get())
    Title = "Кривая Безье 4-ой степени"
    plt.figure("Decart System")
    t = np.arange(0, 1, 0.001)
    x = np.zeros(len(t))
    y = np.zeros(len(t))
    for i in range (len(t)):
```

```

        x[i] = (1 - t[i])**4 * x1 + 4 * (1 - t[i])**3 * t[i] * x2 + 4 *
(1 - t[i])**2 * t[i]**2 * x3 + 4 * (1 - t[i]) * t[i]**3 * x4 + t[i]**4 *
x5
        y[i] = (1 - t[i])**4 * y1 + 4 * (1 - t[i])**3 * t[i] * y2 + 4 *
(1 - t[i])**2 * t[i]**2 * y3 + 4 * (1 - t[i]) * t[i]**3 * y4 + t[i]**4 *
y5
    plt.plot(x, y, 'b')
    plt.scatter(x1, y1, color='red', s = 100, marker='o')
    plt.scatter(x2, y2, color='red', s = 100, marker='o')
    plt.scatter(x3, y3, color='red', s = 100, marker='o')
    plt.scatter(x4, y4, color='red', s = 100, marker='o')
    plt.scatter(x5, y5, color='red', s = 100, marker='o')
    plt.grid(True)
    plt.title(Title)
    plt.xlabel("x")
    plt.ylabel("y")
    plt.show()
window = Tk()
window.title("Кривая Безье 4-ой степени")
window.geometry("400x200")
lb1 = Label(window, text = "Введите координаты точек!")
lb1.place(x = 1, y = 1)
lb2 = Label(window, text = "Координаты первой точки          x1:")
lb2.place(x = 1, y = 25)
txt1 = Entry(window, width = 4)
txt1.place(x = 190, y = 25)
lb3 = Label(window, text = "          y1:")
lb3.place(x = 220, y = 25)
txt2 = Entry(window, width = 4)
txt2.place(x = 255, y = 25)
lb4 = Label(window, text = "Координаты второй точки          x2:")
lb4.place(x = 1, y = 50)
txt3 = Entry(window, width = 4)
txt3.place(x = 190, y = 50)
lb5 = Label(window, text = "          y2:")
lb5.place(x = 220, y = 50)
txt4 = Entry(window, width = 4)
txt4.place(x = 255, y = 50)
lb6 = Label(window, text = "Координаты третьей точки          x3:")
lb6.place(x = 1, y = 75)
txt5 = Entry(window, width = 4)
txt5.place(x = 190, y = 75)
lb7 = Label(window, text = "          y3:")
lb7.place(x = 220, y = 75)
txt6 = Entry(window, width = 4)
txt6.place(x = 255, y = 75)
lb8 = Label(window, text = "Координаты четвертой точки  x4:")
lb8.place(x = 1, y = 100)
txt7 = Entry(window, width = 4)
txt7.place(x = 190, y = 100)
lb9 = Label(window, text = "          y4:")
lb9.place(x = 220, y = 100)
txt8 = Entry(window, width = 4)
txt8.place(x = 255, y = 100)
lb10 = Label(window, text = "Координаты пятой точки          x5:")
lb10.place(x = 1, y = 125)
txt9 = Entry(window, width = 4)
txt9.place(x = 190, y = 125)

```

```
lb11 = Label(window, text = "      y5:")
lb11.place(x = 220, y = 125)
txt10 = Entry(window, width = 4)
txt10.place(x = 255, y = 125)
btn = Button(window, text = "Запуск", bg = "green", fg = "white", command
= clicked)
btn.place(x = 325, y = 150)
window.mainloop()
```

Вывод

В ходе данной лабораторной работы были приобретены навыки по работе с плоскими полиномиальными кривыми. Строить график кривых по заданным точкам и изменять график в зависимости от изменённых точек

Список литературы

1. Работа с библиотекой Matplotlib [Электронный ресурс] URL: <https://matplotlib.org/>
(дата обращения: 07.12.2020)
2. Создание графического интерфейса Tkinter [Электронный ресурс] URL: <https://pythonru.com/uroki/obuchenie-python-gui-uroki-po-tkinter>
(дата обращения: 08.12.2020)