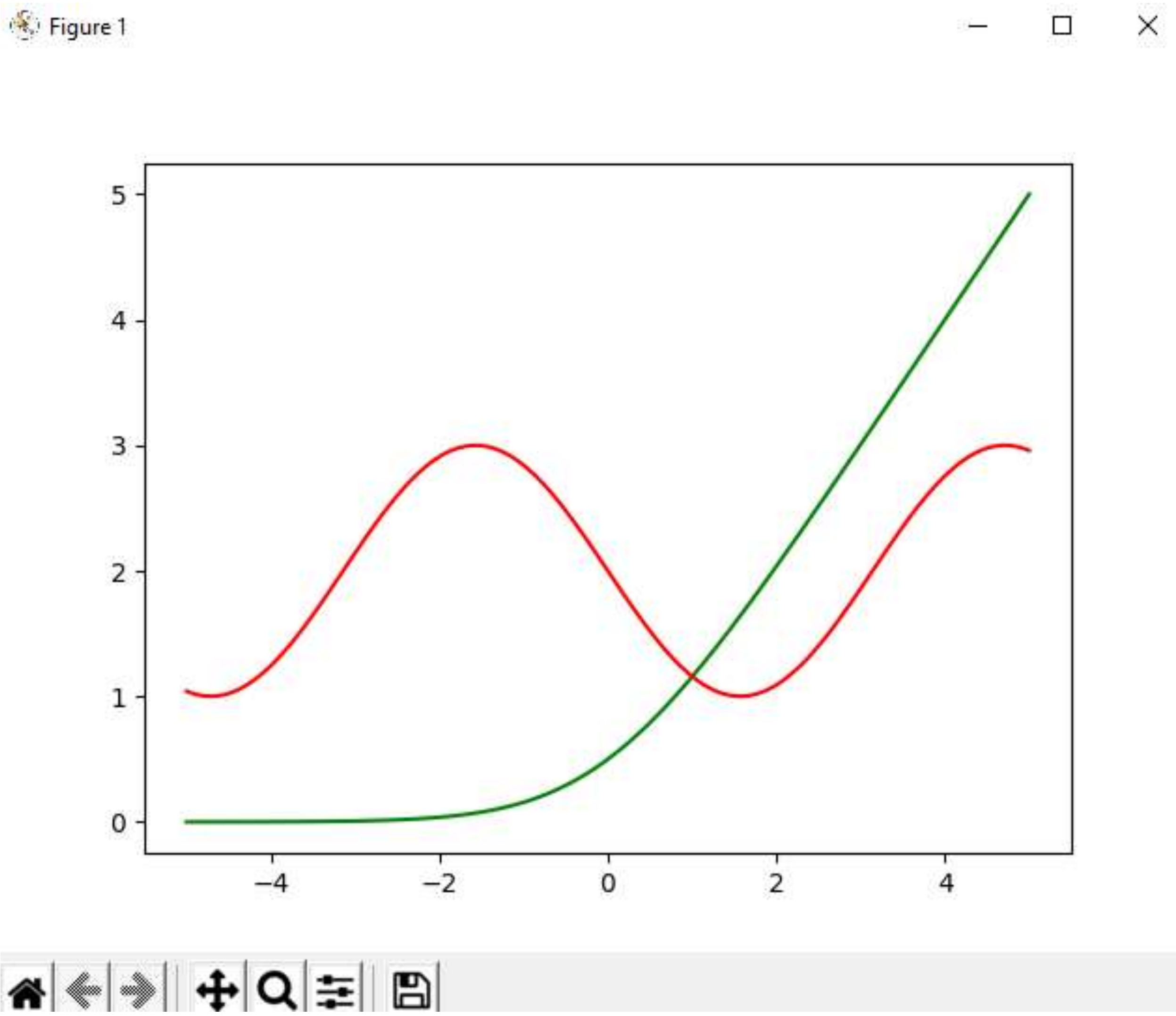
Козунов Алексей, 12 группа, вариант 10

Лабораторная работа 1

Задание 1

10) $\begin{cases} e^{-2x} + \dfrac{x}{y} = 1, \\ \sin x + y = 2; \end{cases}$



Точка пересечения двух графиков (1.002, 1.158):

X принадлежит (0, 1), Y принадлежит (0, 1.6)

Поэтому я взял начальное значения x0, y0 = 0.5, 0.8.

Значение h = [0.000005, 0.000005] было выбрано экспериментальным путем, были выбраны различные h и чем меньше оно становилось, тем точнее становился ответ. Остановился на данном h так как дальнейшее его уменьшение приводит к ухудшению алгоритма.

Козунов Алексей, 12 группа, вариант 10

Дискретный метода Ньютона

```
import math
import matplotlib.pyplot as plt

import numpy as np


def firstFun(_x, _y):
    return math.exp(-2*_x)+_x/_y-1


def secondFun(_x, _y):
    return math.sin(_x)+_y-2


def jacobi(_x, _y, _h1, _h2):
    return [[(firstFun(_x + _h1, _y) - firstFun(_x, _y)) / _h1, (firstFun(_x,
_y + _h2)-firstFun(_x, _y)) / _h2],
            [(secondFun(_x + _h1, _y)-secondFun(_x, _y)) / _h1,
(secondFun(_x, _y + _h2)-secondFun(_x, _y)) / _h2]]


def graph():
    y = lambda x: -x / (np.exp(-2 * x) - 1)
    y1 = lambda x: 2 - np.sin(x)
    fig = plt.subplots()
    x = np.linspace(-5, 5, 100)
    plt.plot(x, y(x), "green")
    plt.plot(x, y1(x), "red")
    plt.show()


if __name__ == '__main__':
    graph()
    h = [0.000005, 0.000005]
    x0, y0 = 0.5, 0.8
    x, y = x0, y0
    xt = [x, y]
    e = 10**(-6)
    i = 0
    while np.sqrt(xt[0]**2 + xt[1]**2) > e:
        i += 1
        print(i)
        v = np.array([-firstFun(x, y), -secondFun(x, y)])
        A = np.array(jacobi(x, y, h[0], h[1]))
        xt = np.linalg.solve(A, v)
        x += xt[0]
        y += xt[1]
    print(x, y)
    print(firstFun(x, y))
    print(secondFun(x, y))
```

Козунов Алексей, 12 группа, вариант 10

Результат:

```
Итерации:
1
2
3
4
5
x, y:
1.0015074076751147 1.1577155156855803
f1:
0.0
f2
0.0
```

Метода Ньютона

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

import numpy as np


def firstFun(_x, _y):
    return math.exp(-2*_x)+_x/_y-1


def secondFun(_x, _y):
    return math.sin(_x)+_y-2


def jacobi(_x, _y):
    return [[-2*math.exp(-2*_x)+1/_y, -_x/(_y**2)],
            [math.cos(_x), 1]]


def graph():
    y = lambda x: -x / (np.exp(-2 * x) - 1)
    y1 = lambda x: 2 - np.sin(x)
    fig = plt.subplots()
    x = np.linspace(-5, 5, 100)
    plt.plot(x, y(x), "green")
    plt.plot(x, y1(x), "red")
    plt.show()


if __name__ == '__main__':
    graph()
    x0, y0 = 0.5, 0.8
    x, y = x0, y0
    e = 10**(-6)
    xt = [x, y]
    i = 0
    print("Итерации:")
    while np.sqrt(xt[0]**2 + xt[1]**2) > e:
```

```
        i += 1
        print(i)
        v = np.array([-firstFun(x, y), -secondFun(x, y)])
        A = np.array(jacobi(x, y))
        xt = np.linalg.solve(A, v)
        x += xt[0]
        y += xt[1]
    print("x, y:")
    print(x, y)
    print("f1:")
    print(firstFun(x, y))
    print("f2")
    print(secondFun(x, y))
```

Результат:

```
Итерации:
1
2
3
4
5
x, y:
1.0015074076751147 1.1577155156855803
f1:
0.0
f2
0.0
```

Вывод:

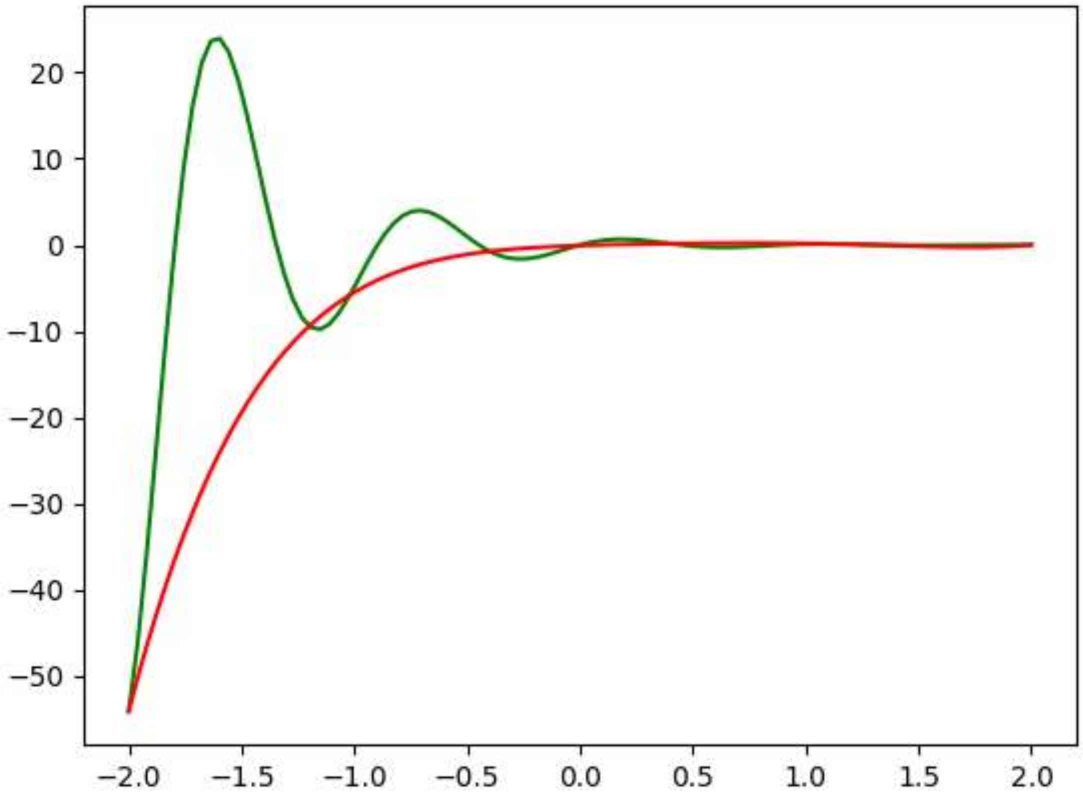
Два этих метода сошлись с ответом за равное количество итераций, но дискретный метод Ньютона без правильно подобранного h работал медленнее и менее точно.

Задание 2

1)

Результат:

На 6:



Время

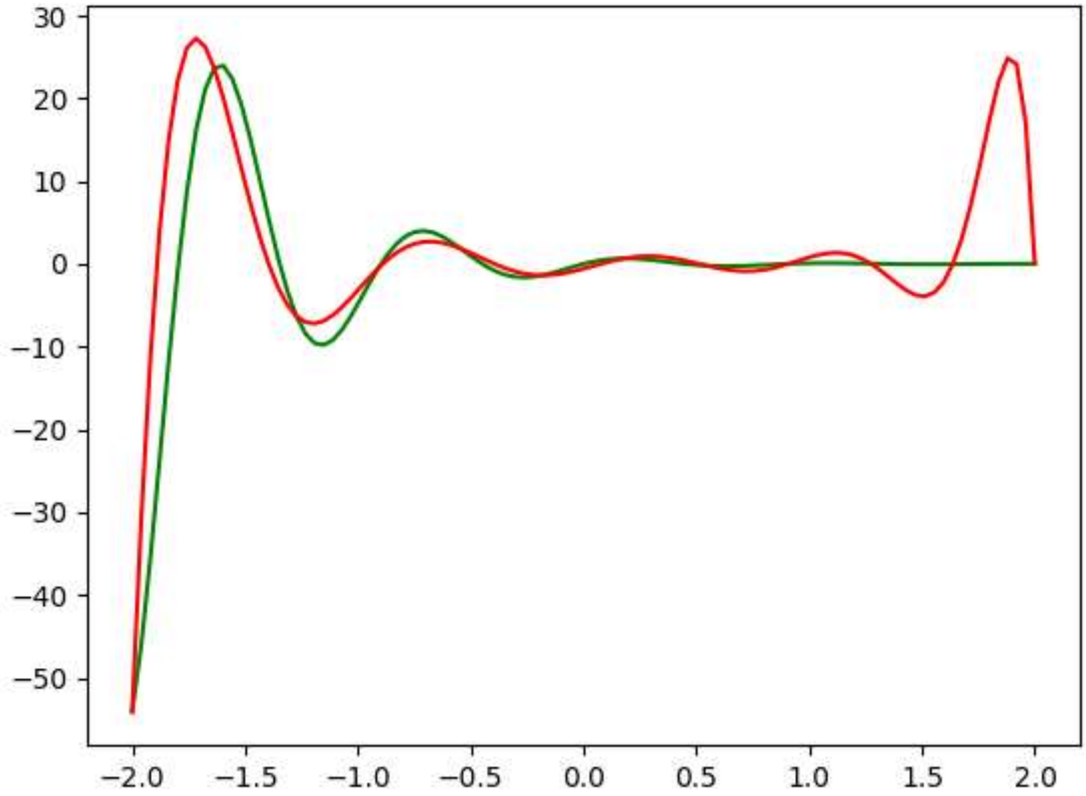
0.0

Многочлен

0.39607645663383 *x^ 5 + -1.3849184285386138 *x^ 4 + 1.5830005895313266 *x^ 3 + -1.2012621840666753 *x^ 2 + 0.8566424949264662 *x^ 1 + -0.06984911889416975 *x^ 0

Козунов Алексей, 12 группа, вариант 10

На 12:



Время

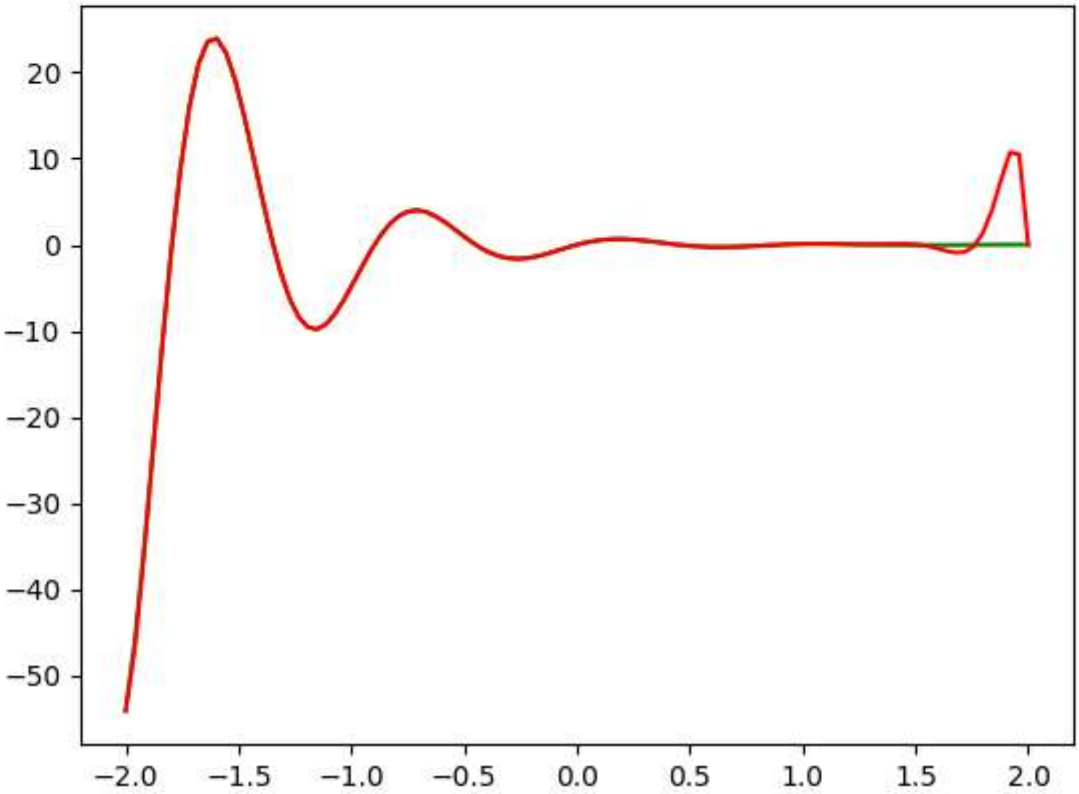
0.0009965

Многочлен

-1.7861073881485101 *x^ 11 + -0.921921382766293 *x^ 10 + 18.88672066472061 *x^ 9 + 5.088049754662393 *x^ 8 + -68.99952544892885 *x^ 7 + -4.624091471178751 *x^ 6 + 102.49858743239899 *x^ 5 + -7.368198881421811 *x^ 4 + -55.96446948940821 *x^ 3 + 7.223020022245112 *x^ 2 + 7.3494512254840565 *x^ 1 + -0.5858777834522018 *x^ 0

Козунов Алексей, 12 группа, вариант 10

На 18:



Время

0.0009965

Многочлен

-0.19279341312590004 *x^ 17 + 0.00472856059873606 *x^ 16 + 3.1658196707041055 *x^ 15 + -0.9802625686210928 *x^ 14 + -21.223950462712445 *x^ 13 + 13.178884960627622 *x^ 12 + 72.36585058181198 *x^ 11 + -70.06888769222283 *x^ 10 + -123.74364353023665 *x^ 9 + 178.2869489653965 *x^ 8 + 77.53563372468165 *x^ 7 + -215.68739386015463 *x^ 6 + 30.76175076860064 *x^ 5 + 107.04178234277438 *x^ 4 + -43.195137249558876 *x^ 3 + -14.15260244348828 *x^ 2 + 7.000327515664893 *x^ 1 + 0.0017464872932050567 *x^ 0

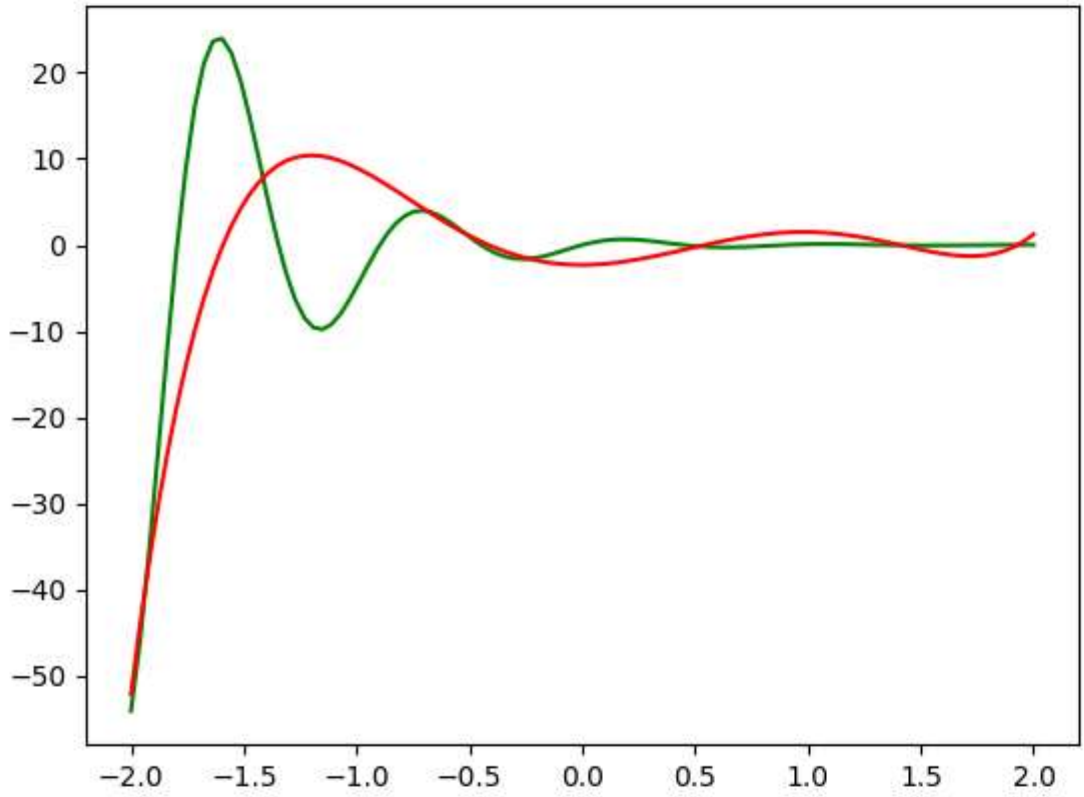
Вывод:

Для решения данной задачи строились интерполяционные многочлены Лагранжа, исходя из результатов видно, что чем больше х-ов мы берем, тем более точное приближение в итоге выходит.

2)

Результат:

На 6:



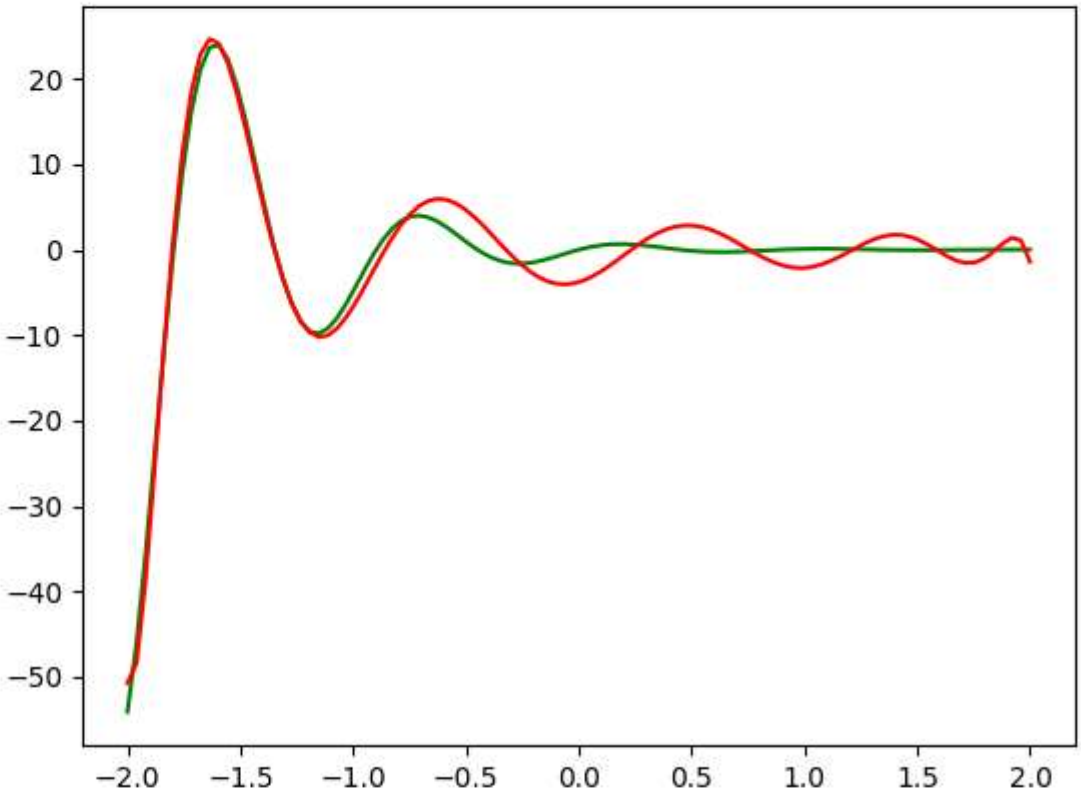
Время

0.0

Многочлен

2.3708500036291853 *x^ 5 + -4.442546531033052 *x^ 4 + -6.163843393501975 *x^ 3 + 11.985904315319333 *x^ 2 + 0.06785858051448454 *x^ 1 + -2.311543444285031 *x^ 0

На 12:



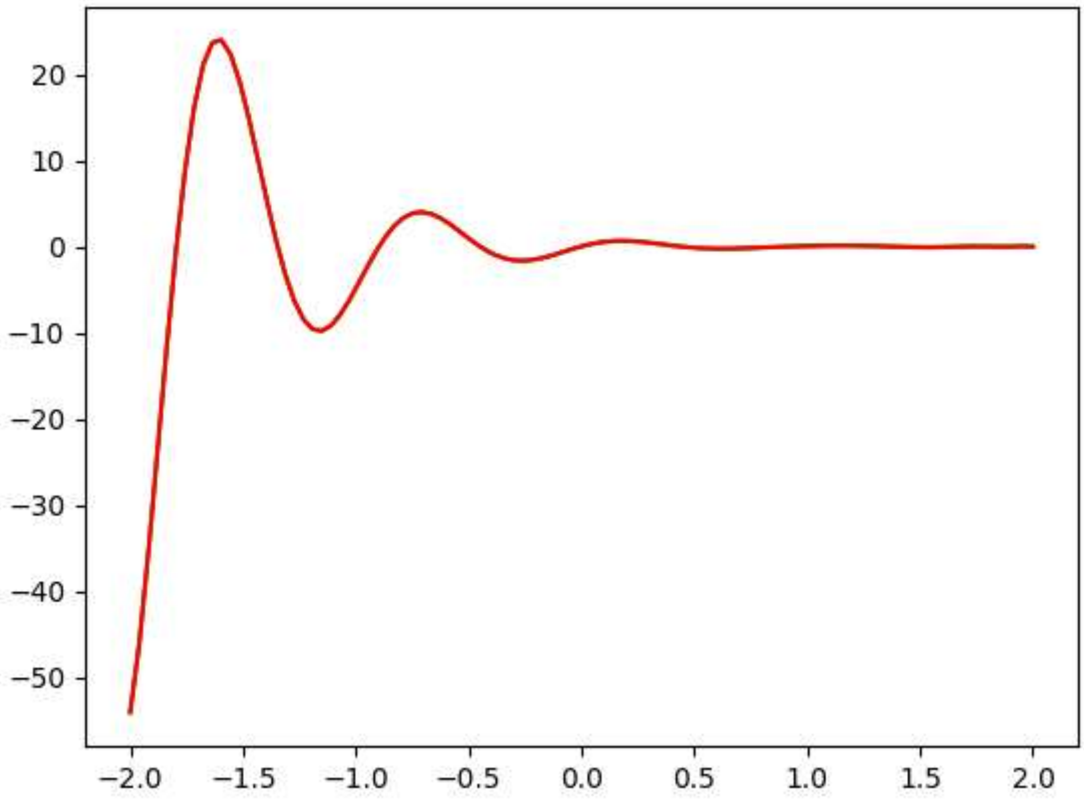
Время

0.0009962

Многочлен

-2.1818358019466957 *x^ 11 + 2.6122101840052903 *x^ 10 + 22.672975000990427 *x^ 9 + -26.75159386277393 *x^ 8 + -81.60467927018277 *x^ 7 + 92.66258150315407 *x^ 6 + 119.59928091920848 *x^ 5 + -125.21943556731338 *x^ 4 + -64.1420873285202 *x^ 3 + 56.088446756666464 *x^ 2 + 7.927473853889397 *x^ 1 + -3.7923219020564165 *x^ 0

На 18:



Время

0.0029918

Многочлен

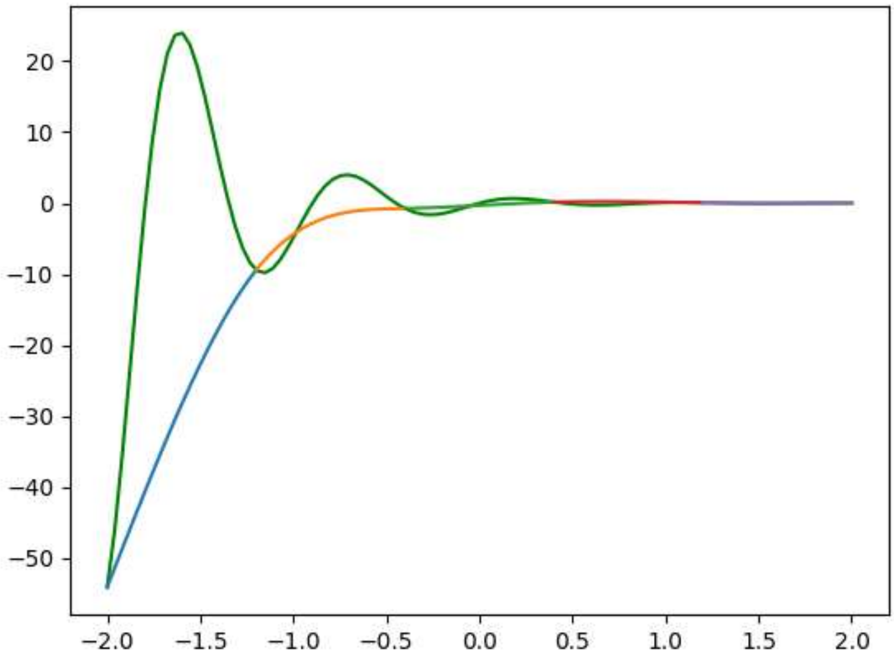
-0.1101258776649035 *x^ 17 + 0.12308052229236022 *x^ 16 + 2.0311339025469364 *x^ 15 + -2.570478688613111 *x^ 14 + -14.981558211517957 *x^ 13 + 21.714056435206473 *x^ 12 + 54.69506821689524 *x^ 11 + -93.57433347768182 *x^ 10 + -96.30587418371387 *x^ 9 + 213.74020568228292 *x^ 8 + 54.57298725586793 *x^ 7 + -244.53114139019038 *x^ 6 + 40.1612667507776 *x^ 5 + 118.55984151983895 *x^ 4 + -44.66760471401568 *x^ 3 + -15.917422330911355 *x^ 2 + 7.041018646617261 *x^ 1 + 0.04683094941855004 *x^ 0

Вывод:

Для решения данной задачи строились интерполяционные многочлены Лагранжа на узлах Чебышёва, исходя из результатов видно, что чем больше x-ов мы берем, тем более точное приближение в итоге выходит.

3)

На 6:



Время

0.0

Сплайн

-9.420394498560837 + 33.07217933955806 *(x- -1.2)+(-85.34620809355087 *(x- -1.2)^2)/2+(-106.68276011693858 *(x- -1.2)^3)/6

-0.7455298387319479 + 0.5248670326625735 *(x- -0.3999999999999999)+(3.9779273263121704 *(x- -0.3999999999999999)^2)/2+(111.6551692748288 *(x- -0.3999999999999999)^3)/6

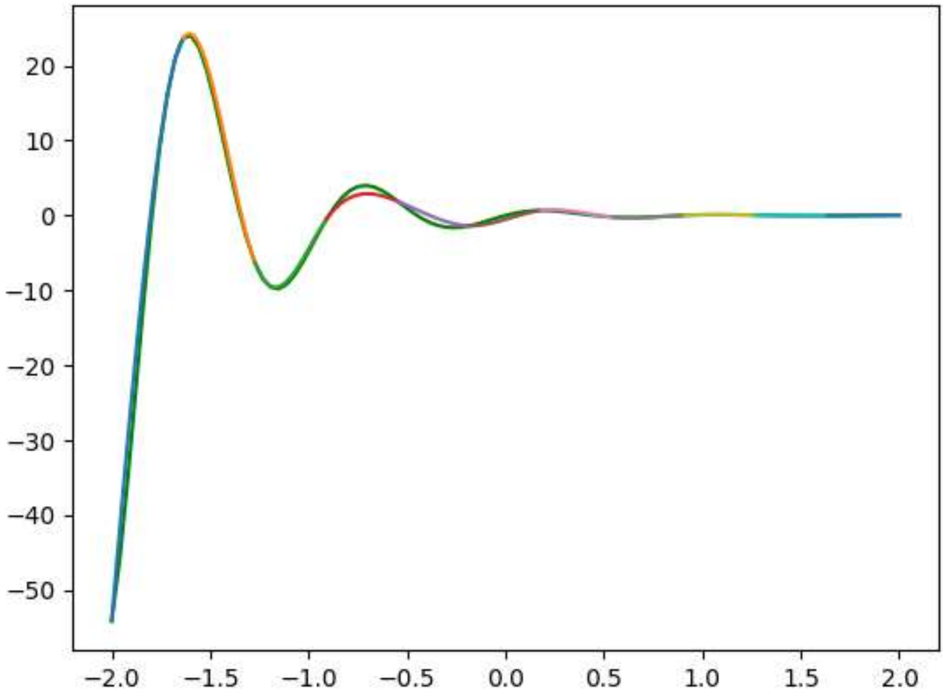
0.15051987850109588 + 0.7192814437739052 *(x- 0.40000000000000036)+(-3.491891298533832 *(x- 0.40000000000000036)^2)/2+(-9.337273281057502 *(x- 0.40000000000000036)^3)/6

0.07752746382513545 + -0.31552792316912937 *(x- 1.2000000000000002)+(0.9048678811762456 *(x- 1.2000000000000002)^2)/2+(5.495948974637597 *(x- 1.2000000000000002)^3)/6

0.018143606607431098 + 0.04641922930136898 *(x- 2.0)+(0 *(x- 2.0)^2)/2+(-1.131084851470307 *(x- 2.0)^3)/6

Козунов Алексей, 12 группа, вариант 10

На 12:



Время

0

Сплайн

23.652766175120217 + 40.82513128394763 *(x- -1.6363636363636362)+(-1426.8757018773358 *(x- -1.6363636363636362)^2)/2+(-3923.9081801626735 *(x- -1.6363636363636362)^3)/6

-6.2869591347370335 + -69.22105057833828 *(x- -1.2727272727272727)+(821.6217016347636 *(x- -1.2727272727272727)^2)/2+(6183.367859658273 *(x- -1.2727272727272727)^3)/6

-0.495096088445911 + 36.839207354985035 *(x- -0.9090909090909092)+(-238.29028300148508 *(x- -0.9090909090909092)^2)/2+(-2914.757957749684 *(x- -0.9090909090909092)^3)/6

1.863997409872487 + -10.890387348573288 *(x- -0.5454545454545454)+(-24.222487868085707 *(x- -0.5454545454545454)^2)/2+(588.6864366168483 *(x- -0.5454545454545454)^3)/6

-1.3751185937900527 + -0.5378436297810527 *(x -0.18181818181818166)+(81.16147832144304 *(x -0.18181818181818166)^2)/2+(289.80590702120406 *(x- -0.18181818181818166)^3)/6

0.6644917941298372 + 3.1458405378206584 *(x- 0.18181818181818166)+(-60.901215399633585 *(x- 0.18181818181818166)^2)/2+(-390.6724077329608 *(x- 0.18181818181818166)^3)/6

-0.21032636089888085 + -2.435982600149417 *(x- 0.5454545454545454)+(30.20118814079814 *(x- 0.5454545454545454)^2)/2+(250.53160973618722 *(x- 0.5454545454545454)^3)/6

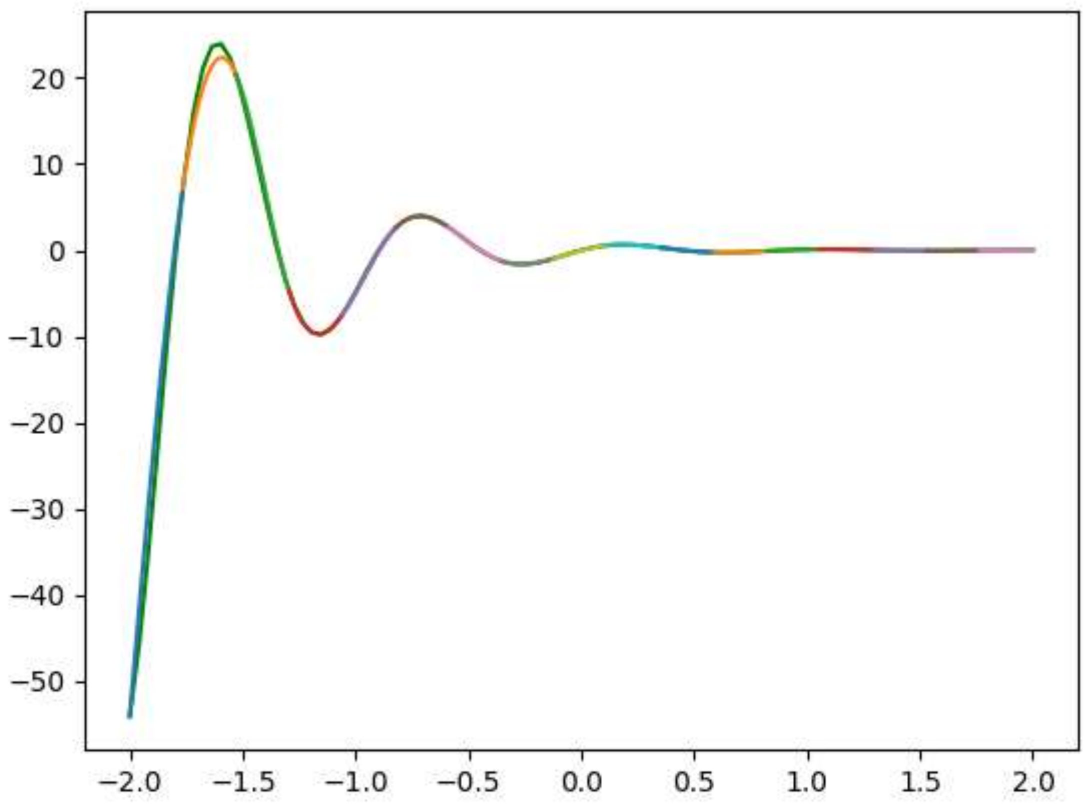
0.013044782239681472 + 1.2236520146832168 *(x- 0.9090909090909092)+(-10.073197759218663 *(x- 0.9090909090909092)^2)/2+(-110.7545612250462 *(x- 0.9090909090909092)^3)/6

0.03868003972072322 + -0.40432265347171814 *(x- 1.272727272727273)+(1.119337084366523 *(x- 1.272727272727273)^2)/2+(30.77947081985926 *(x- 1.272727272727273)^3)/6

-0.033980295232751076 + 0.005681710056087752 *(x- 1.6363636363636367)+(1.1356869150364104 *(x- 1.6363636363636367)^2)/2+(0.04496203434219059 *(x- 1.6363636363636367)^3)/6

0.018143606607431098 + 0.21217024006270765 *(x- 2.0)+(0 *(x- 2.0)^2)/2+(-3.1231390163501285 *(x- 2.0)^3)/6

На 18:



Козунов Алексей, 12 группа, вариант 10

Время

0.0010071

Сплайн

7.223641437857285 + 177.38999040586296 *(x- -1.7647058823529411)+(-1060.4574595611948 *(x- -1.7647058823529411)^2)/2+(-4506.944203135078 *(x- -1.7647058823529411)^3)/6

20.41484724627662 + -61.8324056295328 *(x- -1.5294117647058822)+(-972.9329067396686 *(x- -1.5294117647058822)^2)/2+(371.9793494914863 *(x- -1.5294117647058822)^3)/6

-4.7615245427837775 + -82.87123414090536 *(x- -1.2941176470588236)+(794.102864393001 *(x- -1.2941176470588236)^2)/2+(7509.902027313846 *(x- -1.2941176470588236)^3)/6

-7.512028906251004 + 37.249671248427155 *(x- -1.0588235294117647)+(226.92483141632627 *(x- -1.0588235294117647)^2)/2+(-2410.506640150868 *(x- -1.0588235294117647)^3)/6

2.572792640958315 + 27.385093239908436 *(x- -0.8235294117647058)+(-310.7737444887356 *(x- -0.8235294117647058)^2)/2+(-2285.218947596513 *(x- -0.8235294117647058)^3)/6

2.6860787556942634 + -16.764171518258774 *(x- -0.5882352941176472)+(-64.49500595568576 *(x- -0.5882352941176472)^2)/2+(1046.6846387654618 *(x- -0.5882352941176472)^3)/6

-1.2594839600377352 + -9.189933829573004 *(x- -0.3529411764705883)+(128.87602630951474 *(x- -0.3529411764705883)^2)/2+(821.826887127102 *(x- -0.3529411764705883)^3)/6

-0.9281456459623966 + 7.442545715428357 *(x- -0.11764705882352944)+(12.50004982299677 *(x- -0.11764705882352944)^2)/2+(-494.5979000677014 *(x- -0.11764705882352944)^3)/6

0.5797520031103606 + 2.870009497997787 *(x- 0.11764705882352944)+(-51.36660767115663 *(x- 0.11764705882352944)^2)/2+(-271.433294350152 *(x- 0.11764705882352944)^3)/6

0.3069523901593665 + -3.1750837468670223 *(x- 0.3529411764705883)+(-0.016684910194224606 *(x- 0.3529411764705883)^2)/2+(218.23717173409023 *(x- 0.3529411764705883)^3)/6

-0.255416749015546 + -0.8180761001350034 *(x- 0.5882352941176472)+(20.051249907416388 *(x- 0.5882352941176472)^2)/2+(85.2887229748451 *(x- 0.5882352941176472)^3)/6

-0.09545257922995323 + 1.3167247876932162 *(x- 0.8235294117647056)+(-1.9054423608764937 *(x- 0.8235294117647056)^2)/2+(-93.31594214024474 *(x- 0.8235294117647056)^3)/6

0.10874076548073479 + 0.19418525918971885 *(x- 1.0588235294117645)+(-7.636143631403245 *(x- 1.0588235294117645)^2)/2+(-24.355480399738692 *(x- 1.0588235294117645)^3)/6

0.026892638563538802 + -0.5335642975850678 *(x- 1.2941176470588234)+(1.450272398817564 *(x- 1.2941176470588234)^2)/2+(38.61726812843844 *(x- 1.2941176470588234)^3)/6

-0.04498686968775497 + -0.019955417247691065 *(x- 1.5294117647058822)+(2.9154030840501384 *(x- 1.5294117647058822)^2)/2+(6.22680541223844 *(x- 1.5294117647058822)^3)/6

-0.006210807561530777 + 0.19131702848119467 *(x- 1.7647058823529411)+(-1.1195872953546104 *(x- 1.7647058823529411)^2)/2+(-17.148709112470183 *(x- 1.7647058823529411)^3)/6

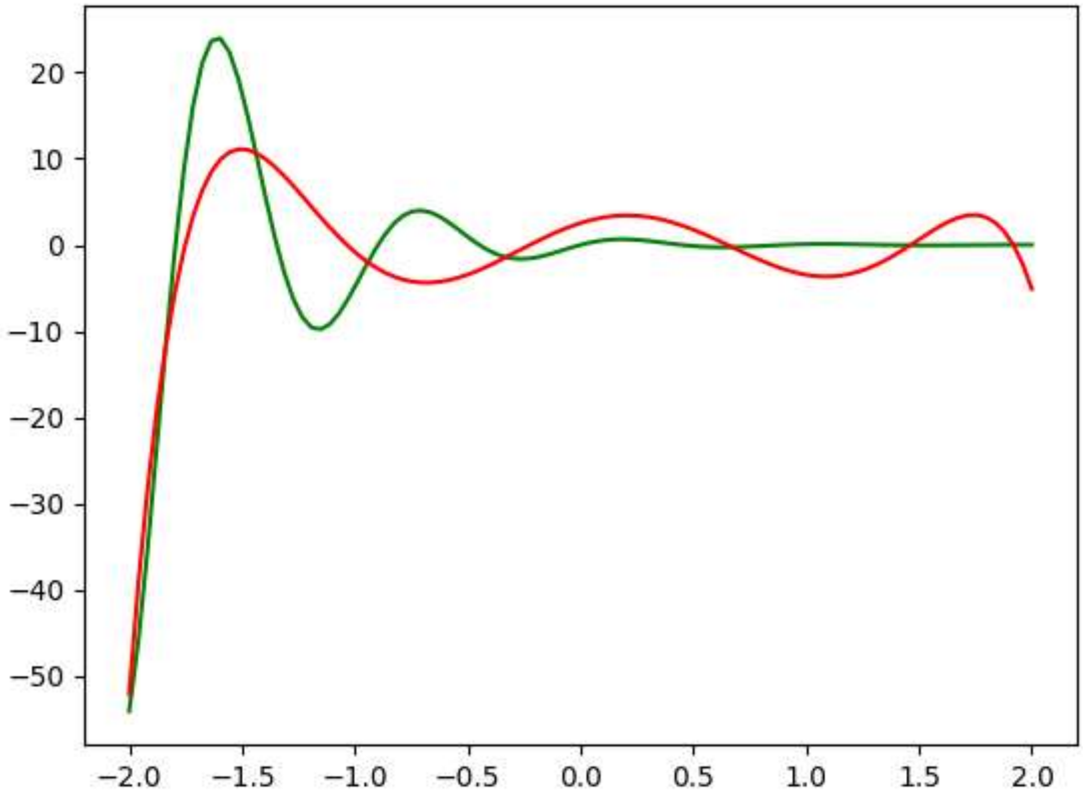
0.018143606607431098 + 0.05960087608653463 *(x- 2.0)+(0 *(x- 2.0)^2)/2+(4.758246005257094 *(x- 2.0)^3)/6

Вывод:

Для решения данной задачи строится трёхмерный сплайн на n узлах, исходя из результатов видно, что чем больше узлов мы берем, тем более точное приближение в итоге выходит.

4)

Для 6:



Время

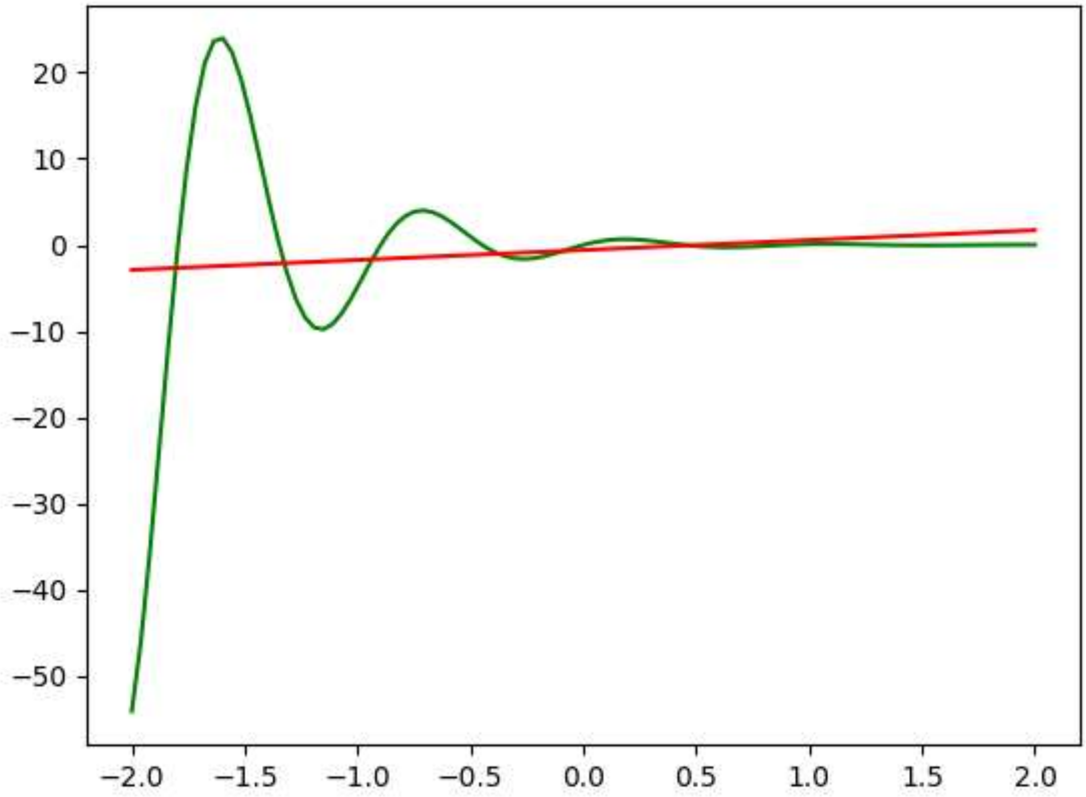
0.0029916

Многочлен

2.4350081478731 64 *x^ 0 + 7.637854804972815 *x^ 1 + -19.43992803336418 *x^ 2 + -9.245468165153664 *x^ 3 + 18.199669772164206 *x^ 4 + 2.2494555717574496 *x^ 5 + -3.966754270382547 *x^ 6

Козунов Алексей, 12 группа, вариант 10

Для 1:



Время

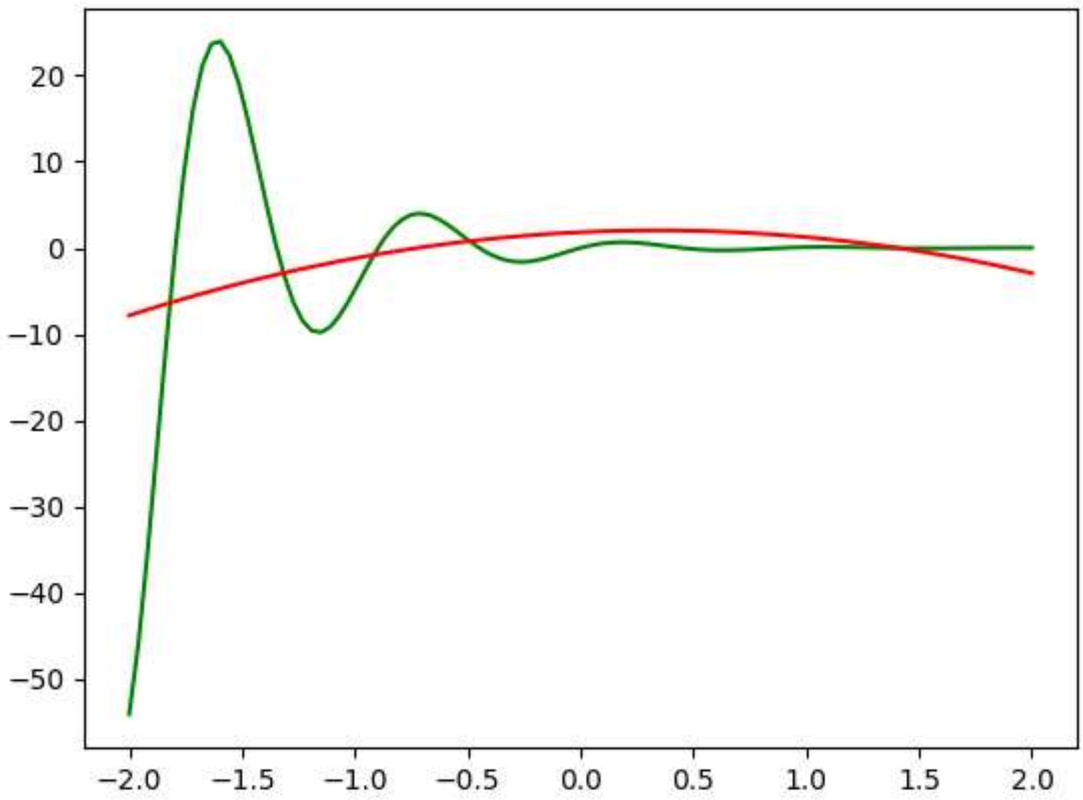
0.0009976

Многочлен

-0.9057448746736785 *x^ 0 + 1.2544854561568086 *x^ 1

Козунов Алексей, 12 группа, вариант 10

Для 2:



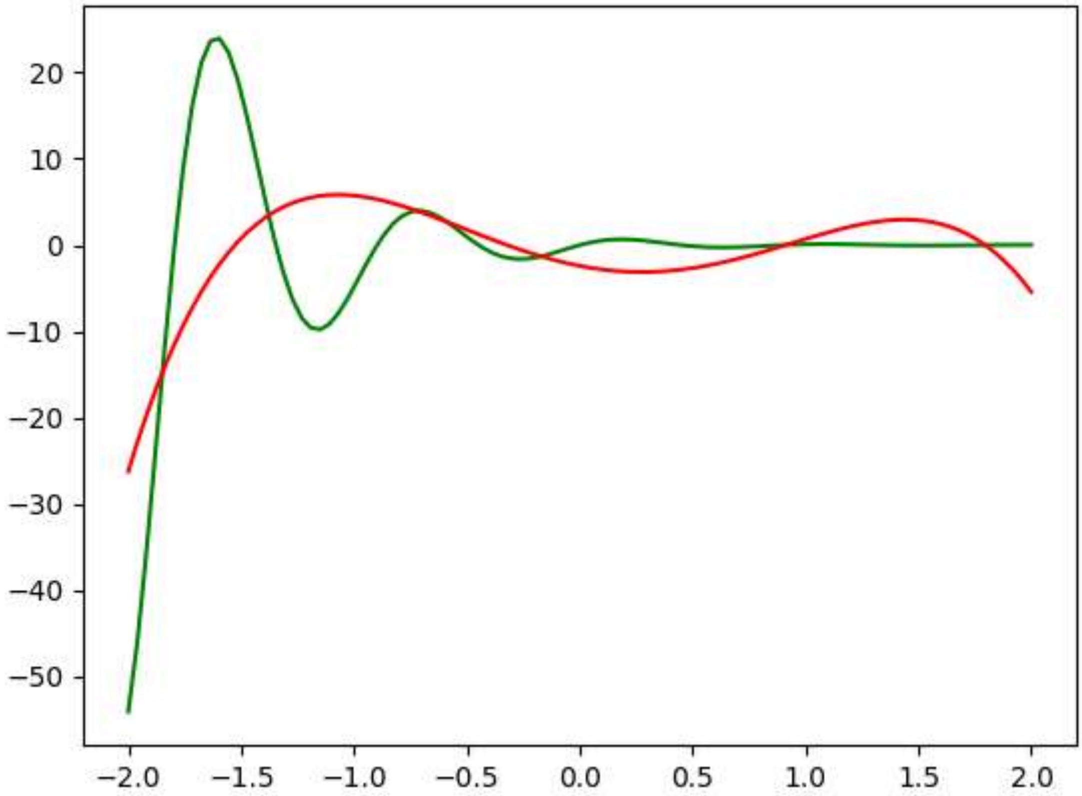
Время

0.0009963

Многочлен

-0.08502771118287912 *x^ 0 + -1.8462858180968236 *x^ 1 + 1.1834365285943915 *x^ 2

Для 4:



Время

0.0019952

Многочлен

-1.5058768708300747 *x^ 0 + -1.9382220672234898 *x^ 1 + 5.573343746627696 *x^ 2 + 0.1514189435735661 *x^ 3 + -1.3385105473026617 *x^ 4

Вывод:

Для решения данной задачи было построено среднеквадратичное приближение, исходя из результатов видно, что чем больше узлов мы берем, тем более точное приближение в итоге выходит.

Итоговый вывод:

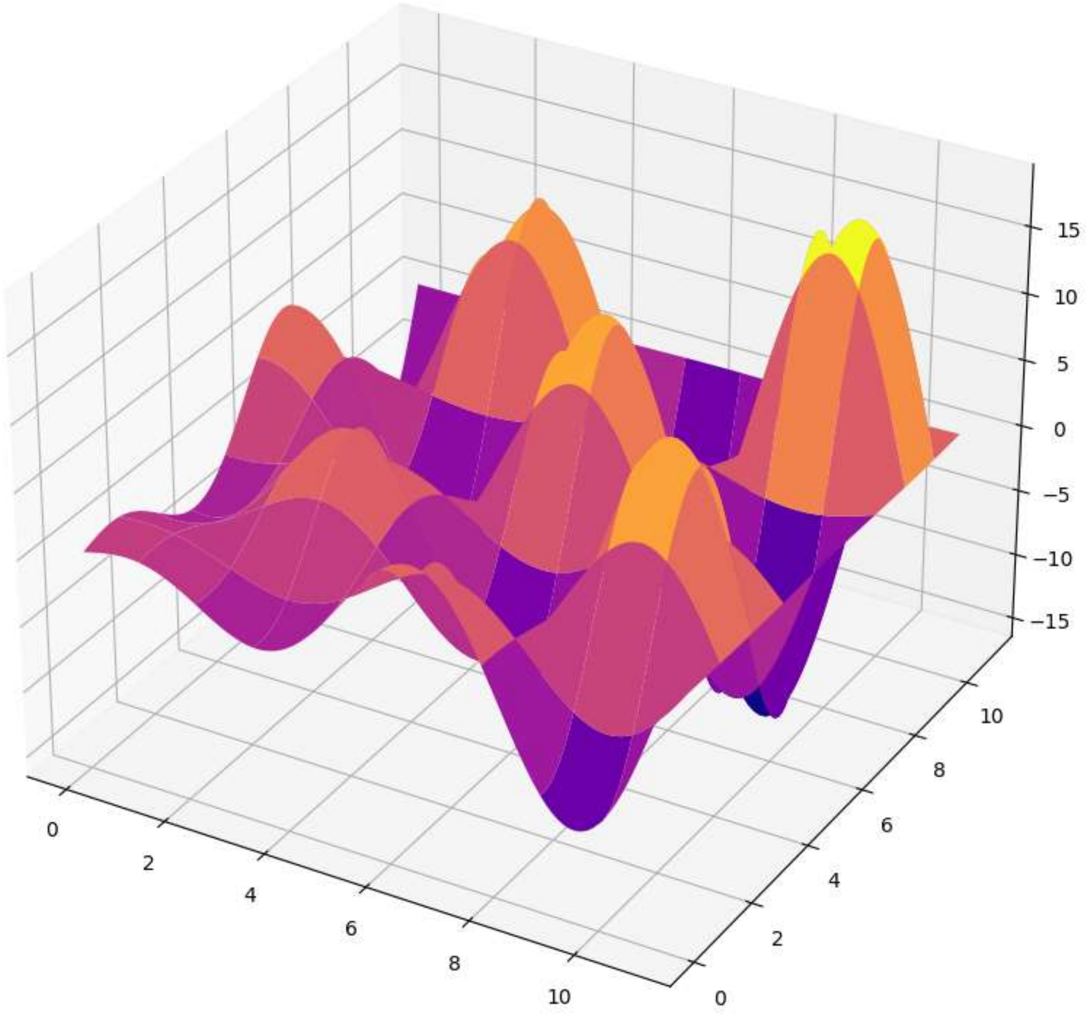
Так как при замере времени очень сильно влияло на сколько загружен компьютер, то выбрать лучший алгоритм исходя от времени выполнения не удастся. Однако исходя из графиков можно сказать, что на 18 узлах лучше всего приближает интерполяционный многочлен Лагранжа на узлах Чебышёва.

Козунов Алексей, 12 группа, вариант 10

Задание 3

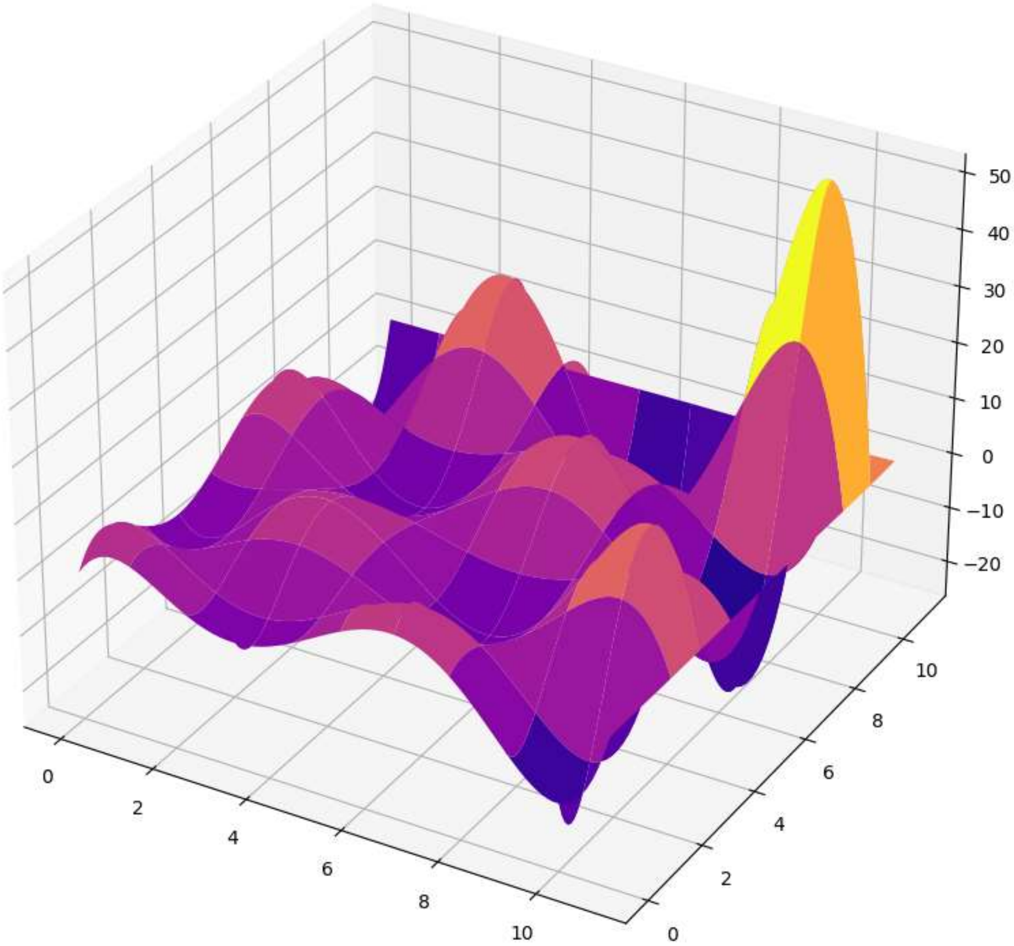
1)

Исходный график:



Козунов Алексей, 12 группа, вариант 10

На 6х6:

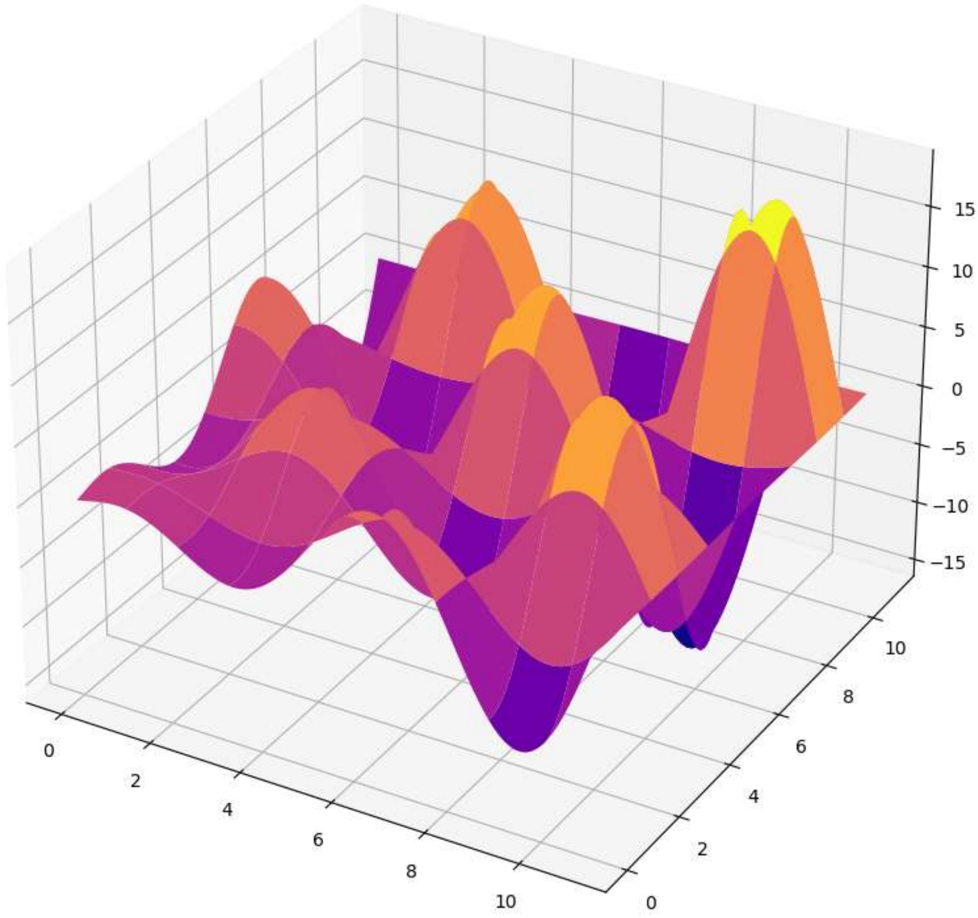


Время

9.8434414 сек

Козунов Алексей, 12 группа, вариант 10

На 12x12:

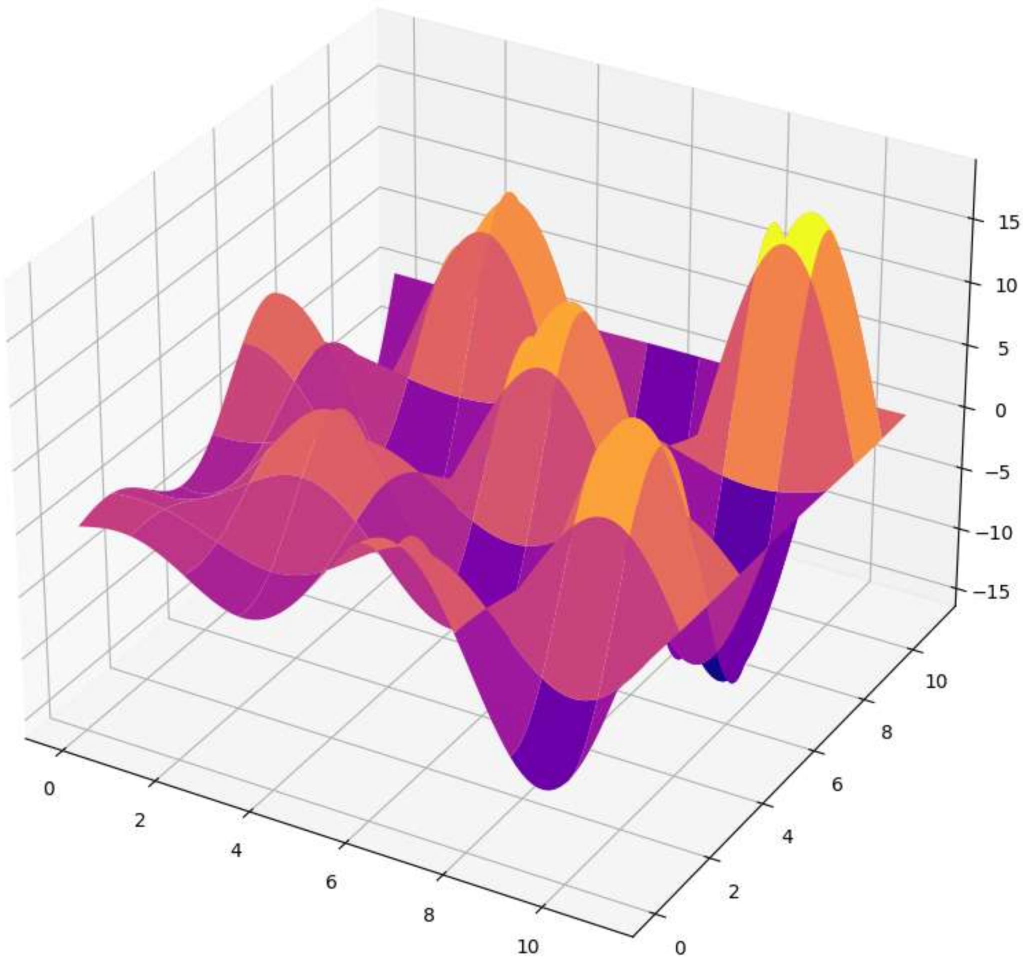


Время

8.9019612 сек

Козунов Алексей, 12 группа, вариант 10

На 18x18:



Время

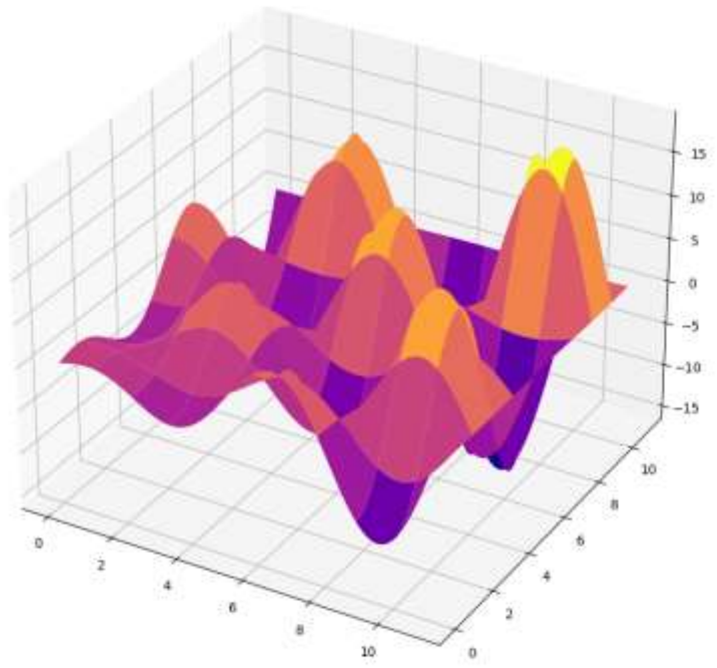
32.3617809 сек

Для решения данной задачи были построены интерполяционные многочлены двух переменных, исходя из результатов видно, что чем больше узлов мы берем, тем более точное приближение в итоге выходит.

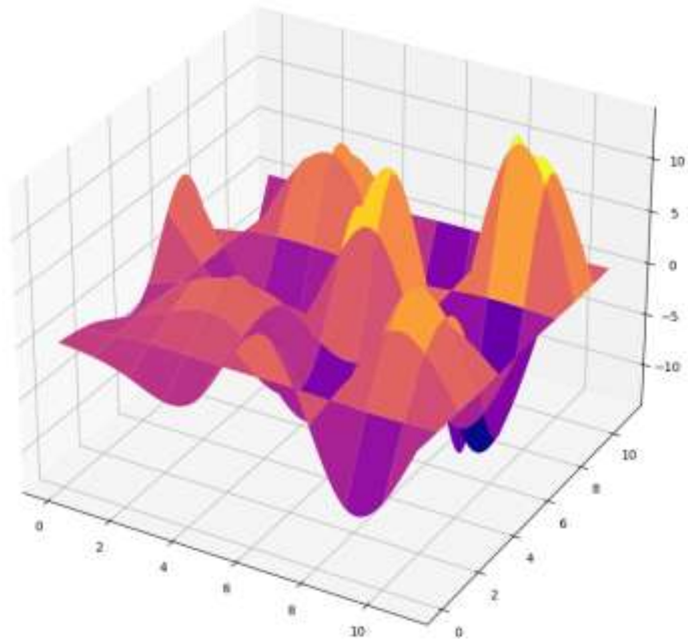
Козунов Алексей, 12 группа, вариант 10

2)

Исходный график



На 6х6:



Время

0.0009972 сек

Альфа

[[[1.5238676860474776, 1.1937138665205056, -4.9210738897763315, 5.250620594706195, -0.034379772379954626], [1.2422103999876979, -2.876193782941969, 1.474008037474983, 2.5123071714770613, -4.859562926933634], [-0.05312433721258447, -3.690879465450749, 7.645608393102523, -6.701700089464271, 0], [-0.024147426005720212, -1.6535250582900747, 5.152949026615124, -6.521503855712179, 3.0462273133928504]], [[1.1937138665205054, 0.8311907381287138, -3.212414805970118, 3.2904371785694004, -0.02094650195061061], [0.9151586072419198, -1.9615104178635596, 0.9225257835432892, 1.6249251570186096, -3.0702241788638585], [-0.33014228832230996, -2.2850113708644897, 4.906862463052534, -4.268317578074972, 0], [-0.15006467651014088, -0.8885768557009908, 3.269033560871374, -4.170536382330684, 1.9401443536704415]], [[-4.9210738897763315, -3.212414805970118, 11.918834120426968, -11.869196062897455, 0.07401578747923618], [-3.6533442094753346, 7.48537007696581, -3.3246251753834093, -5.991571089423986, 11.138884533605195], [1.9557345470187317, 8.170369349745945, -17.99763776097251, 15.57314147548108, 0], [0.888970248644878, 2.824834001239642, -11.894548686690205, 15.259445107478903, -7.078700670673218]], [[5.250620594706195, 3.2904371785694004, -11.869196062897455, 11.578545043289344, -0.07107512386500342], [3.8215656023706455, -7.601930322268728, 3.240951153608382, 5.940090977907844, -10.912968330195525], [-2.4674347891942237, -7.917561505932479, 17.774726484002578, -15.320963007366704, 0], [-1.1215612678155562, -2.477330325790116, 11.678312722697752, -15.043495223349673, 6.964074094257592]], [[-0.034379772379954626, -0.020946501950610607, 0.07401578747923618, -0.07107512386500342, 0.00043090965899160894], [-0.024689041870299043, 0.04809625378025672, -0.01988384617001311, -0.03691808080164899, 0.06721315416718476], [0.01781811454577812, 0.048350336045636204, -0.11015042690951787, 0.09466475906257617, 0], [0.00809914297535369, 0.013878282499935491, -0.07204580134325185, 0.09309781180549728, -0.04302943593753462]]]

Бэта

[[[1.2422103999876979, 0.9151586072419199, -3.6533442094753346, 3.8215656023706455, -0.024689041870299043], [0.9803230543348749, -2.1820599353581143, 1.0721604014661743, 1.8567005955780433, -3.550796646135121], [-0.20415080972022004, -2.670742817273406, 5.629124941659123, -4.915906583375605, 0], [-0.09279582260010001, -1.1211781852514482, 3.772667163151149, -4.793196147743058, 2.2345029924434567]], [[-2.876193782941969, -1.9615104178635603, 7.48537007696581, -7.601930322268728, 0.04809625378025672], [-2.182059935358114, 4.610547819590154, -2.1307260794918994, -3.7791251894477864, 7.105489805666382], [0.9098795899638787, 5.265218369080001, -11.393649186427323, 9.89510454101288, 0], [0.41358163180176305, 1.9796994450527827, -7.572212525230602, 9.676706239745545, -4.4977747913694905]], [[1.4740080374749835, 0.9225257835432892, -3.3246251753834093, 3.240951153608382, -0.01988384617001311], [1.0721604014661734, -2.1307260794918985, 0.9071531716945214, 1.6636025337116656, -3.0550978576138306], [-0.6960145261835, -2.2157004565056564, 4.977408866675129, -4.289727628477725, 0], [-0.31637023917431817, -0.6907663319646165, 3.2695951469003566, -4.212334770524024, 1.9498761947626022]], [[2.5123071714770613, 1.6249251570186096, -5.991571089423986, 5.940090977907844, -0.03691808080164899], [1.8567005955780438, -3.779125189447784, 1.663602533711665, 3.009032083086438, -5.579749490663328], [-1.040311259146124, -4.083166727240991, 9.031101021022309, -7.807983248863426, 0], [-0.4728687541573291, -1.383116121861303, 5.9610307946651355, -7.654129213584424, 3.5490832949379207]], [[-4.859562926933634, -3.0702241788638585, 11.138884533605195, -10.912968330195525, 0.06721315416718476], [-3.5507966461351206, 7.105489805666384, -3.0550978576138306, -5.57974949066333, 10.276362121033515], [2.2146462457594285, 7.47288689224194, -16.709784767951223, 14.414646919724406, 0], [1.0066573844361038, 2.3901093847647776, -10.992123481905983, 14.147468948943468, -6.552112236238366]]]

Гамма

[[[-0.05312433721258458, -0.3301422883223101, 1.9557345470187317, -2.4674347891942237, 0.01781811454577812], [-0.20415080972022015, 0.9098795899638786, -0.6960145261835, -1.0403112591461245, 2.2146462457594276], [-0.7993906563970006, 1.8121455652007266, -3.272049307152889, 2.9590522771868666, 0], [-0.36335938927136385, 1.1870619189080576, -2.310997669251643, 2.832318901972616, -1.3450237623576666]], [[-3.6908794654507484, -2.2850113708644897, 8.170369349745945, -7.917561505932479, 0.048350336045636204], [-2.670742817273406, 5.265218369080002, -2.2157004565056573, -4.083166727240993, 7.472886892241939], [1.8121455652007266, 5.402364604211463, -12.203199900198426, 10.505503290439028, 0], [0.8237025296366938, 1.63191774500488, -8.002529320186312, 10.322137813926116, -4.7752287683813766]], [[7.645608393102523, 4.906862463052534, -17.99763776097251, 17.774726484002578, -

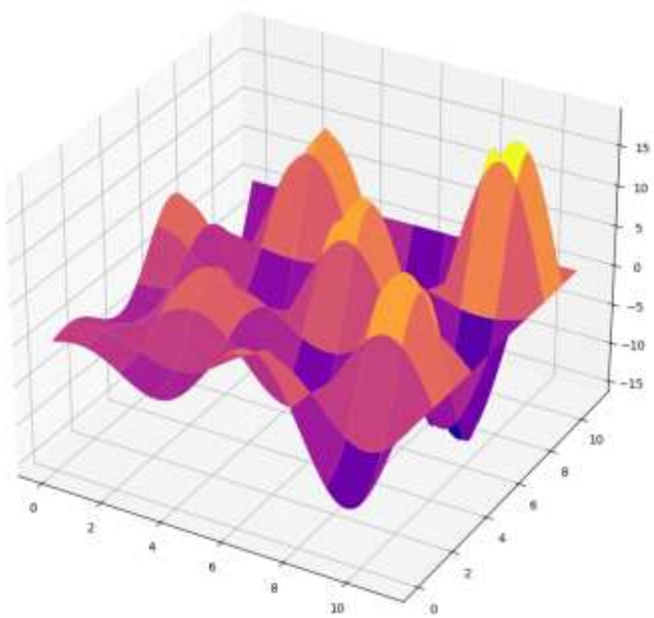
0.11015042690951787], [5.629124941659121, -11.393649186427325, 4.977408866675129, 9.031101021022309, -16.709784767951223], [-3.2720493071528898, -12.203199900198424, 27.08597994847338, -23.40080526270322, 0], [-1.4872951396149499, -4.059613905929788, 17.858718113032637, -22.948538732352997, 10.636729664865099]], [[-6.701700089464271, -4.268317578074972, 15.57314147548108, -15.320963007366704, 0.09466475906257617], [-4.915906583375605, 9.895104541012882, -4.289727628477726, -7.807983248863429, 14.414646919724408], [2.959052277186867, 10.50550329043903, -23.40080526270322, 20.202391062352582, 0], [1.3450237623576666, 3.4302050060237104, -15.411958433246477, 19.81963469320718, -9.182905028342082]], [[0, 0, 0, 0, 0], [0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0], [0.0, 0.0, 0.0, 0.0, 0.0]]]

Дэльта

[[[-0.024147426005720264, -0.15006467651014094, 0.888970248644878, -1.1215612678155562, 0.00809914297535369], [-0.09279582260010005, 0.41358163180176305, -0.3163702391743182, -0.47286875415732926, 1.0066573844361035], [-0.36335938927136385, 0.8237025296366939, -1.4872951396149494, 1.3450237623576666, 0], [-0.16516335875971083, 0.5395735995036626, -1.0504534860234742, 1.2874176827148254, -0.611374437435303]], [[-1.6535250582900745, -0.8885768557009908, 2.824834001239642, -2.477330325790116, 0.013878282499935491], [-1.121178185251448, 1.9796994450527832, -0.6907663319646169, -1.3831161218613028, 2.390109384764778], [1.1870619189080578, 1.6319177450048798, -4.059613905929789, 3.430205006023711, 0], [0.5395735995036626, 0.20220719368037365, -2.58705984133394, 3.4044631417970446, -1.559184093647141]], [[5.152949026615123, 3.269033560871374, -11.894548686690205, 11.678312722697752, -0.07204580134325185], [3.772667163151149, -7.5722125252306025, 3.2695951469003575, 5.961030794665137, -10.992123481905981], [-2.310997669251642, -8.002529320186312, 17.858718113032637, -15.411958433246475, 0], [-1.0504534860234735, -2.5870598413339407, 11.755112469644976, -15.123034793763232, 7.00543565147567]], [[-6.521503855712179, -4.170536382330684, 15.259445107478903, -15.043495223349673, 0.09309781180549728], [-4.793196147743057, 9.676706239745549, -4.2123347705240235, -7.654129213584425, 14.14746894894347], [2.8323189019726156, 10.322137813926116, -22.948538732353, 19.819634693207185, 0], [1.2874176827148252, 3.404463141797045, -15.123034793763235, 19.440078829800083, -9.008924860548719]], [[3.0462273133928504, 1.9401443536704415, -

7.078700670673218, 6.964074094257592, -0.04302943593753462], [2.2345029924434567, -4.497774791369491, 1.9498761947626022, 3.549083294937922, -6.552112236238367], [-1.3450237623576664, -4.775228768381377, 10.636729664865099, -9.182905028342082, 0], [-0.6113744374353028, -1.5591840936471413, 7.00543565147567, -9.008924860548717, 4.174047740155491]]]

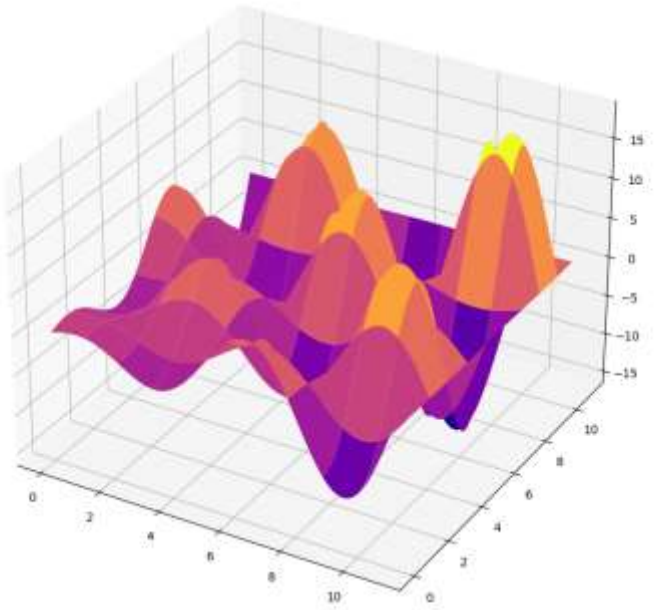
На 12x12:



Время

0.0009996 сек

На 18x18:



Время

0.0049856 сек

Для решения данной задачи был построен бикубический сплайн, исходя из результатов видно, что чем больше узлов мы берем, тем более точное приближение в итоге выходит.

Итоговый вывод:

Исходя из выше приведенных данных видим, что что бикубический сплайн работает лучше.