Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования

«Алтайский государственный технический университет им. И. И. Ползунова»

Факультет информационных технологий Кафедра прикладной математики

Отчет защищен с оценкой

Преподаватель

(подпись)

«\_ » 2023 г.

Отчет

По лабораторной работе №3

по дисциплине «Вычислительные алгоритмы»

Студент группы ПИ-02 Чередов Р.А.

Преподаватель Проскурин А.В.

Барнаул 2023

# Задание

Составить программу для построения интерполяционного многочлена Лагранжа (Ньютона). Программа должна работать в двух режимах:

а) по заданной таблице значений функции определять приближенное значение функции в некоторой точке, вводимой пользователем;

б) по заданной аналитически функции y = f (x) и массиву значений аргумента (массив читается из файла) вычислить таблицу значений функции. Используя полученную таблицу, построить интерполяционный многочлен после чего нарисовать графики функции y = f (x) и интерполяционного многочлена. Найти величину максимального отклонения заданной функции от интерполирующей в точках, в которых вычислялись значения функций для построения графиков (для изображения графиков число точек, в которых вычисляются функции должно быть существенно больше, чем число узлов интерполяции).

Исследовать путем проведения вычислительных экспериментов влияние количества и расположения узлов интерполирования, участков интерполирования на величину погрешности интерполирования. В качестве функций, для которых проводится анализ, помимо придуманных Вами функций рекомендуется рассмотреть y = |x| при |x| <= 1, y = e

-x при |x| <= 4, y = sinx при |x| <= Pi.

# Решение

1. **Код программы (Python 3)**

mport numpy as np

mport matplotl b.pyplot as plt from numpy mport \*

from os mport system

def lagrange\_ nterpolat on(x, y, x\_ nterp): """

Вычисляет интерполяционный многочлен Лагранжа. Аргументы:

1. -- массив узлов интерполяции
2. -- массив значений функции в узлах интерполяции

x\_ nterp -- точка, в которой требуется вычислить значение интерполяционного многочлена

Возвращает:

Значение интерполяционного многочлена в точке x\_ nterp """

Вычисляем количество узлов интерполяции n = len(x)

Инициализируем массив базисных многочленов Лагранжа lj = np.ones(n)

Вычисляем базисные многочлены Лагранжа for n range(n):

for j n range(n): f != j:

lj[ ] \*= (x\_ nterp - x[j]) / (x[ ] - x[j])

Вычисляем значение интерполяционного многочлена Лагранжа y\_ nterp = np.sum(y \* lj)

return y\_ nterp

def read\_f le\_x(f lename): """

Функция для чтения массива значений аргумента из файла. """

w th open(f lename, 'r') as f:

x = [float(l ne) for l ne n f] return np.array(x)

def read\_f le\_xy(f lename): """

Функция для чтения значений x, y из файла. """

x = []

y = []

w th open(f lename, 'r') as f: for l ne n f:

data = l ne.spl t() x.append(float(data[0])) y.append(float(data[1]))

return np.array(x), np.array(y)

def plot\_graphs(x, y, x\_ nterp, y\_ nterp, x\_ ntersect, y\_ ntersect): """

Функция для построения графиков и вычисления максимального отклонения. """

Строим график исходной функции plt.plot(x, y, label='Исходная функция')

Строим график интерполяционного многочлена

plt.plot(x\_ nterp, y\_ nterp, label='Интерполяционный многочлен')

Отмечаем точки пересечения

plt.plot(x\_ ntersect, y\_ ntersect, 'o', color='red', label='Узлы')

Определяем максимальное отклонение max\_err = max\_error(y\_ nterp, y)

pr nt('Максимальное отклонение: ', max\_err)

Добавляем легенду и выводим графики на экран plt.legend()

plt.show()

def max\_error(y1, y2): """

Функция для вычисления максимального отклонения между двумя массивами. """

max\_err = 0

for n range(len(y1)):

err = abs(y1[ ] - y2[ ]) f err > max\_err:

max\_err = err return max\_err

wh le (True):

\_ = system('cls')

Выбираем режим работы программы

mode = nput('Выберите режим работы программы (a/b): ') f mode == 'a':

Считываем данные из файла

f lename = nput('Введите имя файла с массивом значений аргумента: ') x1 = read\_f le\_x(f lename)

Запрашиваем ввод функции f(x) func = nput('Введите выражение: ')

Вычисляем значения функции y1 = []

for x n x1:

y1.append(eval(func))

Создаем массивы с множеством точек для построения графика функции y = f(x) x\_forgraph = np.arange(x1[0], x1[-1], 0.001)

y\_forgraph = []

for x n x\_forgraph: y\_forgraph.append(eval(func))

Строим интерполяционный многочлен

x\_ nterp = np.arange(x1[0], x1[-1], 0.001) y\_ nterp = []

for x n x\_ nterp:

y\_ nterp.append(lagrange\_ nterpolat on(x1, y1, x ))

Строим графики

plot\_graphs(x\_forgraph, y\_forgraph, x\_ nterp, y\_ nterp, x1, y1) el f mode == 'b':

Считываем данные из файла

f lename = nput('Введите имя файла со значениями x, y: ')

Задаем таблицу точек

x2, y2 = read\_f le\_xy(f lename)

Задаем точку, в которой нужно вычислить значение многочлена x0 = float( nput('Введите значение точки: '))

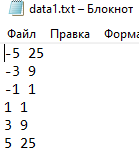
Строим интерполяционный многочлен

y0 = lagrange\_ nterpolat on(x2, y2, x0)

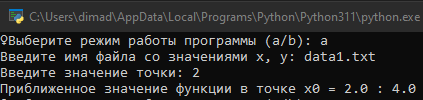
pr nt('Приближенное значение функции в точке x0 =', x0, ':', y0) else:

pr nt('Некорректный ввод.')

# Первый режим работы

* 1. Тест №1 Исходные данные:

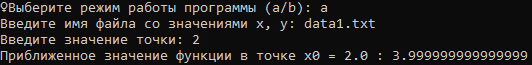
Результат:



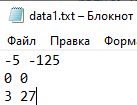
* 1. Тест №2 Исходные данные:



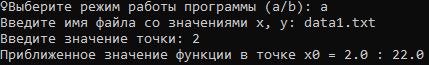
Результат:



* 1. Тест №3 Исходные данные:



Результат:



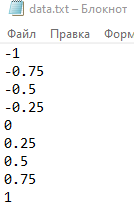
# Второй режим работы

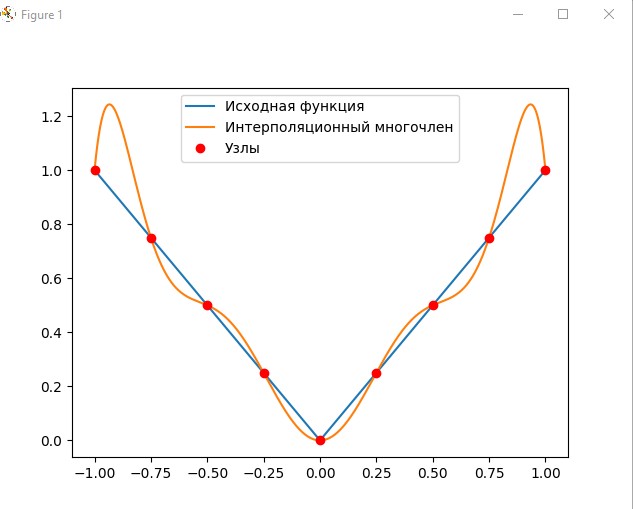
**3.1** | | | |

Исходные данные:

Результат:

Тест №1

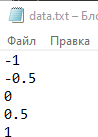




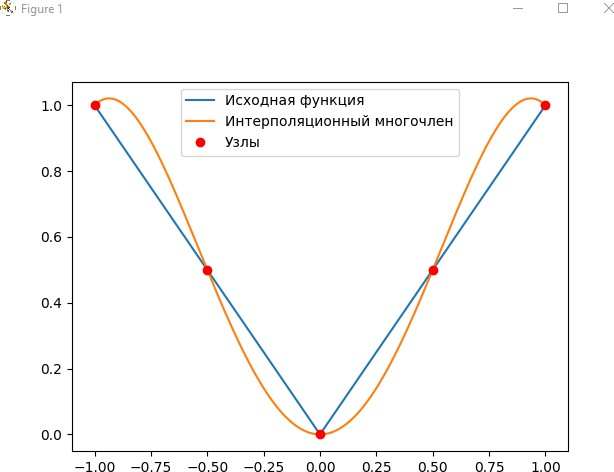


Тест №2

Исходные данные:



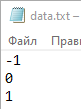
Результат:



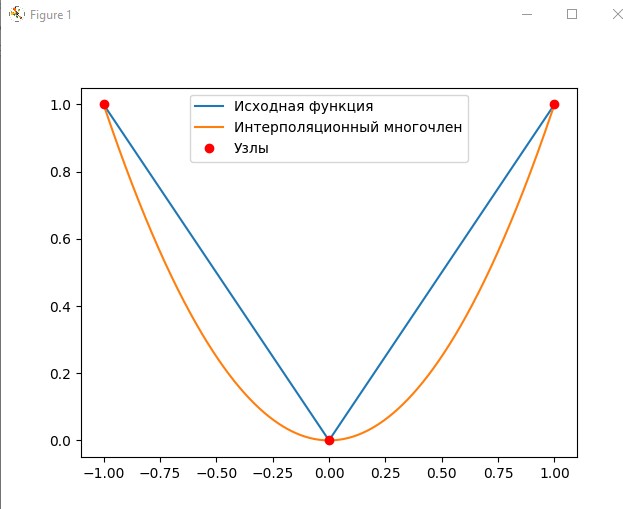


Тест №3

Исходные данные:



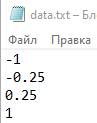
Результат:



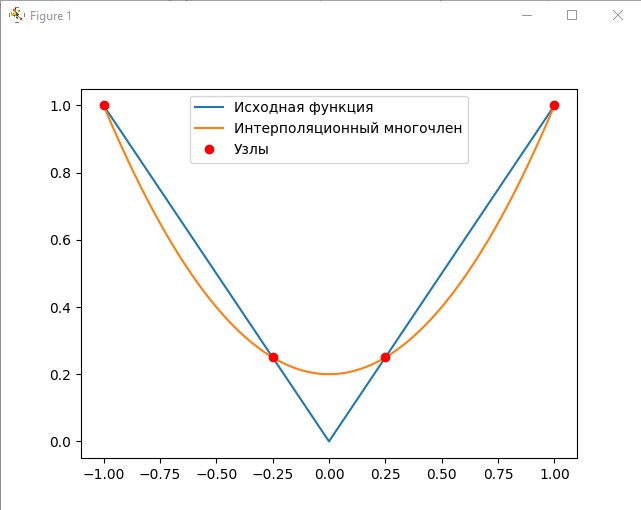


Тест №4

Исходные данные:



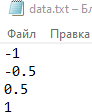
Результат:



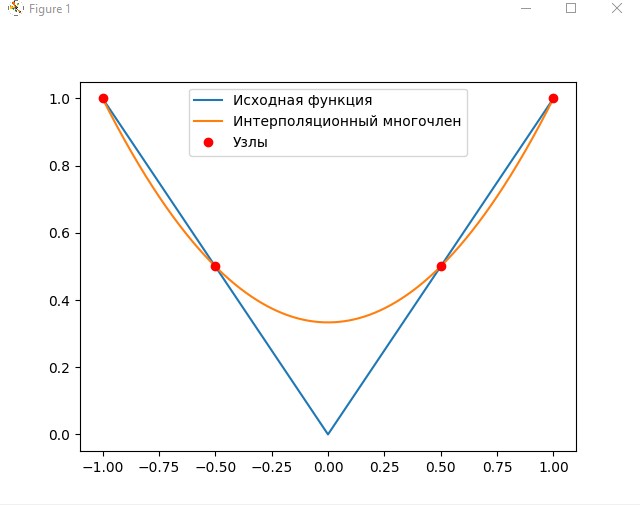


Тест №5

Исходные данные:



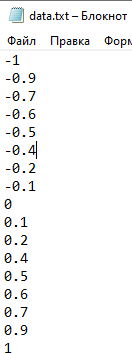
Результат:



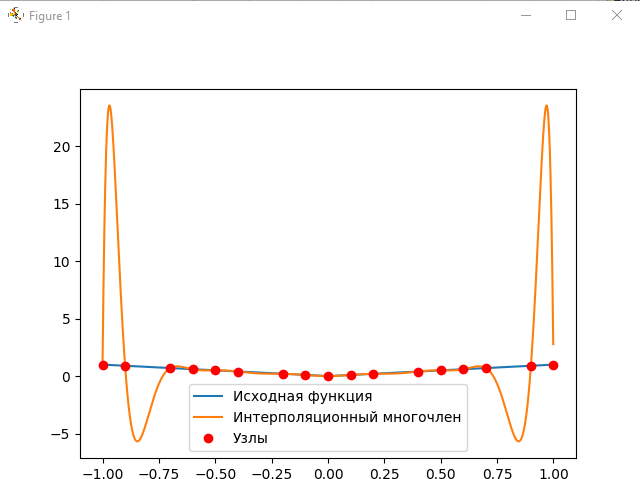


Тест №6

Исходные данные:



Результат:



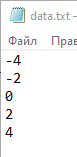


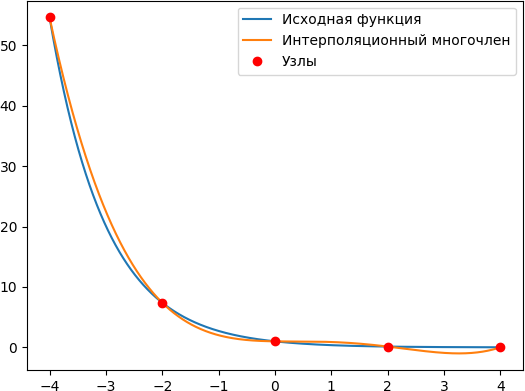
**3.2** | |

Исходные данные:

Результат:

Тест №1

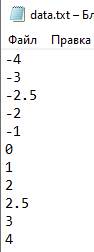




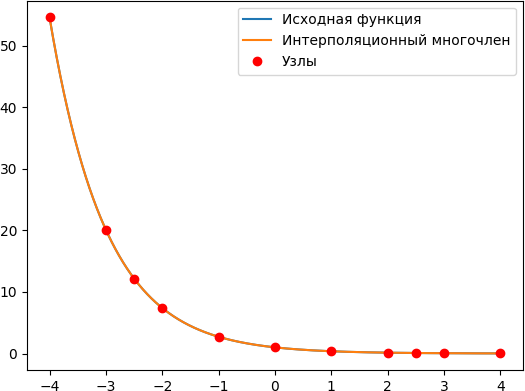


Тест №2

Исходные данные:

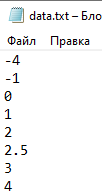


Результат:

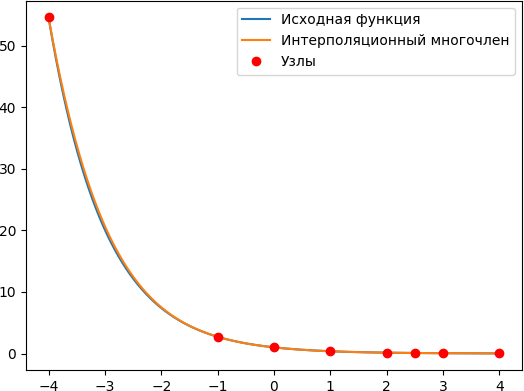


Тест №3

Исходные данные:



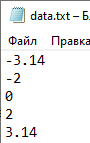
Результат:



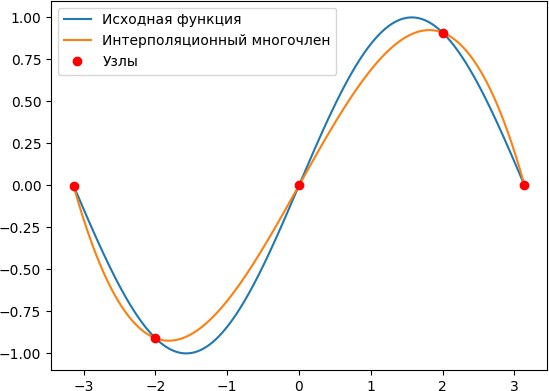
# 3.3 | |

Исходные данные:

Тест №1

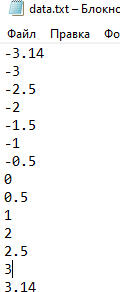


Результат:

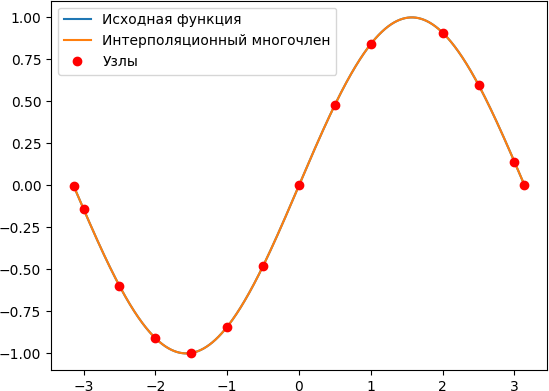


Тест №2

Исходные данные:



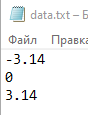
Результат:





Тест №3

Исходные данные:



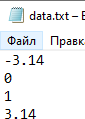
Результат:



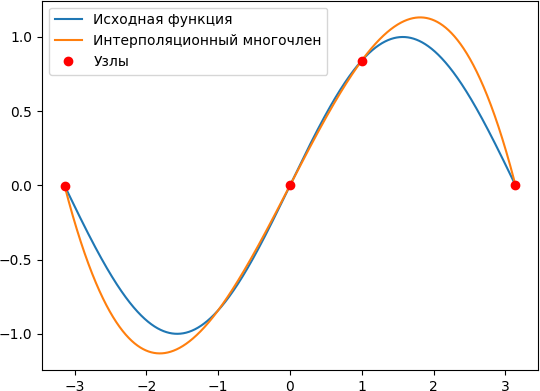


Тест №3

Исходные данные:



Результат:





# Вывод

Для различных функций увеличение или уменьшение точек интерполяции по разному влияло на результат и максимальное отклонение. Например, для функции:

| | | |

увеличение точек интерполяции привело к значительному увеличению максимального отклонения, а для функции:

| |

наоборот, уменьшило величину максимального отклонения.

Изменение расположения точек интерполирования уменьшило максимальное отклонение на одном участке, но увеличило на другом.