БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет прикладной математики и информатики

**Отчет**

**Методы численного анализа**

Лабораторная работа 2

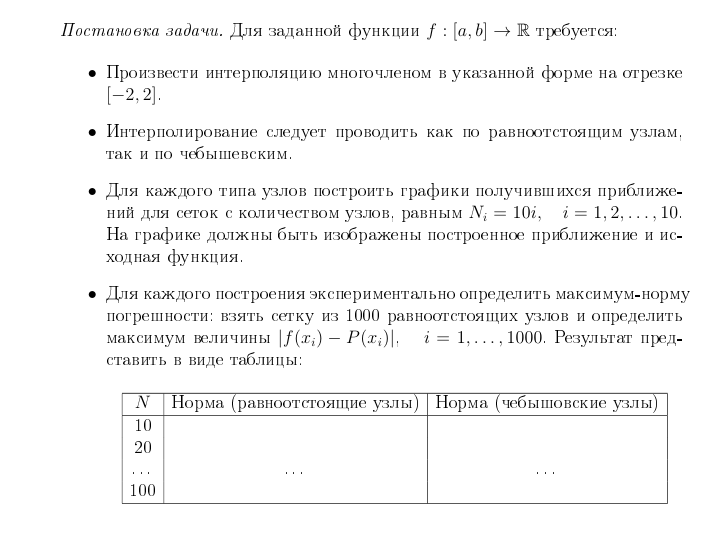
**Выполнила**

Юрковская Екатерина Артуровна

Студентка 2 курса 3 группы

Минск 2020

1. **Постановка задачи.**

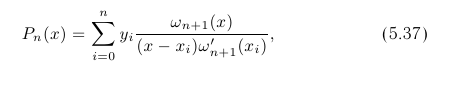


Примечание: , ИМ в барицентрической форме

1. **Теоретические сведения**

*2.1. Многочлен Лагранжа в барицентрической форме.*

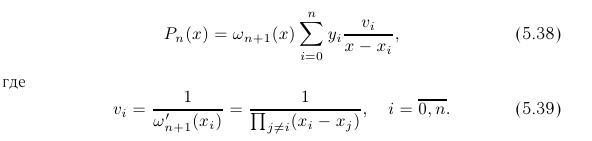
Запишем многочлен Лагранжа в виде:



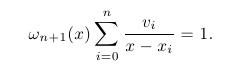


Чтобы избежать громоздких умножений в каждом слагаемом формулы

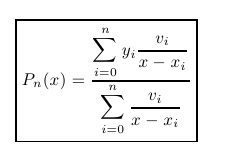
(5.37), достаточно просто вынести общий множитель и записать в следующем виде:



Формула (5.38) называется первой формой барицентрической интерполяционной формулы. Существует вторая, еще более эффективная форма записи барицентрической формулы. Строится она так: строим по формуле (5.38) интерполяционный многочлен для f(x) = 1, который, очевидно, тождественно равен 1. То есть,



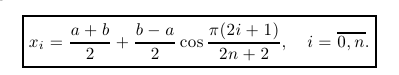
Тогда P(x) можно записать в виде



Эту формулу называют просто барицентрической интерполяционной формулой. [1]

2.2) Интерполирование по чебышевским узлам.

Для вычисления узлов интерполирования будем использовать формулу [2] :



1. **Исходный код программы.**

import numpy as np

import math

import matplotlib.pyplot as plt

from pylab import \*

#исходная функция

func\_x = np.linspace(-2.0, 2.0, 1000)

func\_y = [math.cos(x)\* math.cos(x) - x for x in func\_x]

# вычисление интерполяционного полинома в барицентрической форме

def bar\_form(x, y, val):

v=[]

for i in range(len(x)):

x\_=1.0

for j in range(len(x)):

if i == j:

continue

x\_\*= x[i]-x[j]

v.append(1.0/x\_)

chisl = 0; znam = 0

for i in range(len(y)):

koef=v[i]/(val-x[i])

znam+=koef

chisl+=y[i]\*koef

return chisl/znam

# интерполяция функции по узлам Чебышева

def interpolate\_cheb(a, b, nodes\_cnt):

nodes\_x = np.asarray([0.5 \* (a + b) + 0.5 \* (b - a) \* cos(pi \* (2 \* k - 1) / (2 \* nodes\_cnt))

for k in range(1, nodes\_cnt + 1)], dtype = np.float64)

nodes\_y = np.asarray([math.cos(x)\* math.cos(x) - x for x in nodes\_x], dtype = np.float64)

x = np.linspace(a, b, 1000)

y = [bar\_form(nodes\_x, nodes\_y, x\_val) for x\_val in x]

return x, y

# интерполяция функции по равноотстоящим узлам

def interpolate(a, b, nodes\_cnt):

nodes\_x = np.linspace(a, b, nodes\_cnt)

nodes\_y = [math.cos(x)\* math.cos(x) - x for x in nodes\_x]

x = np.linspace(a,b, 1000)

y = [bar\_form(nodes\_x, nodes\_y, x\_val) for x\_val in x]

return x, y

# построение графика для равноотстоящих узлов

def draw\_plot(x, y, n):

fig = plt.figure(figsize=(15,10))

ax = plt.subplot(111)

ax.plot(func\_x, func\_y, 'r-', label = 'f(x)')

ax.plot(x, y, 'b--', label = 'polynom')

plt.grid(True)

plt.title('n = {}'.format(n))

ax.legend()

show()

# построение графика для чебышевских узлов

def draw\_plot\_cheb(x, y, n):

fig = plt.figure(figsize=(15,10))

ax = plt.subplot(111)

ax.plot(func\_x, func\_y, 'r-', label = 'f(x)')

ax.plot(x, y, 'g--', label = 'polynom\_cheb')

plt.grid(True)

plt.title('n = {}'.format(n))

ax.legend()

show()

n=10

x1 = []; y1 = []

x1, y1 = interpolate (-2.0, 2.0, n)

x2 = []; y2 = []

x2, y2 = interpolate\_cheb (-2.0, 2.0,n)

draw\_plot(x1,y1,n)

draw\_plot\_cheb (x2,y2,n)

max\_n1 = 0

max\_n2 = 0

for i in range(1000):

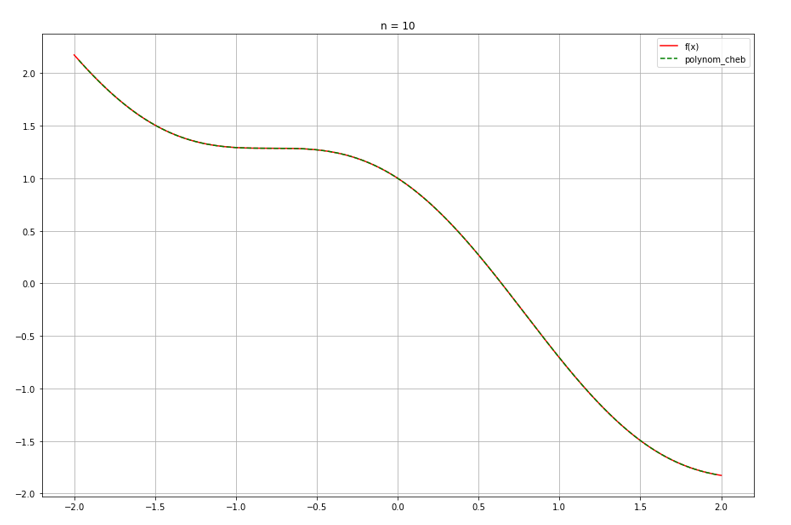
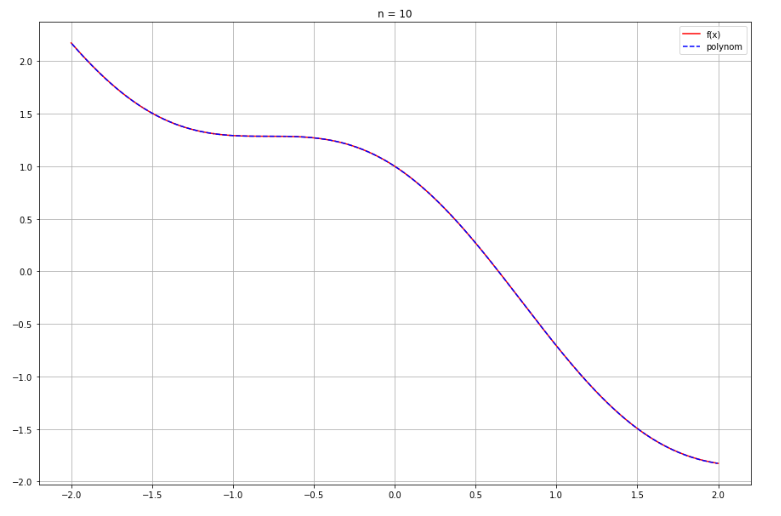
max\_n1 = max(max\_n1, abs(y1[i] - func\_y[i]))

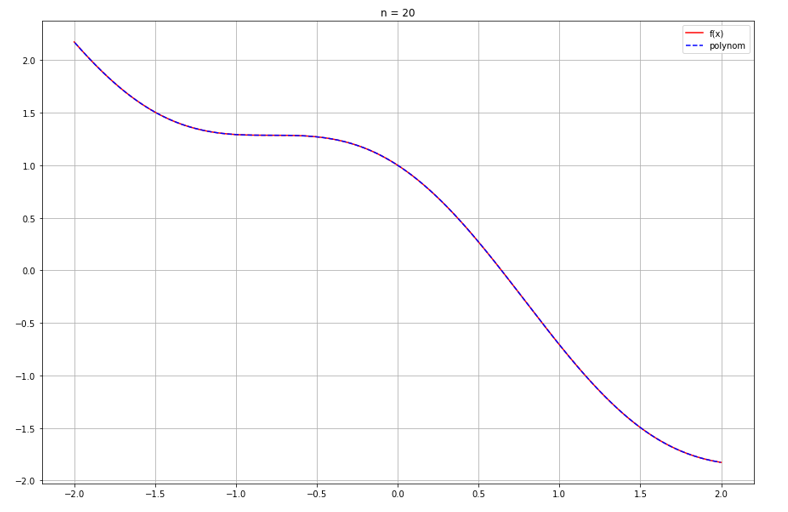
max\_n2 = max(max\_n2, abs(y2[i] - func\_y[i]))

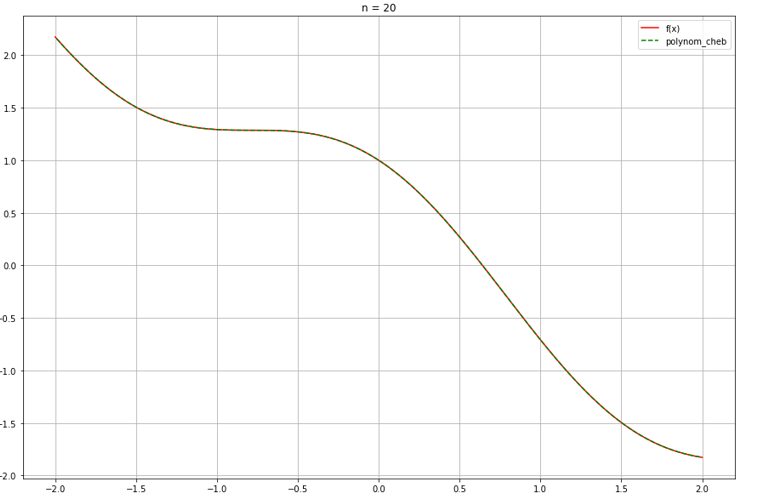
print(max\_n1)

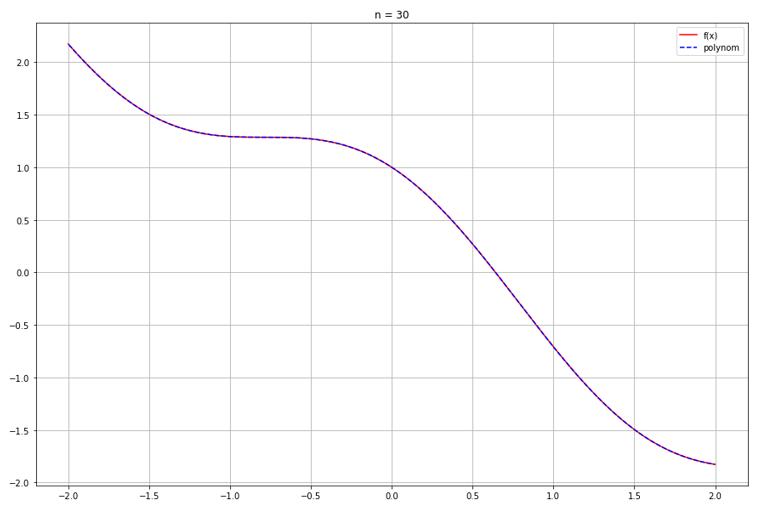
print(max\_n2)

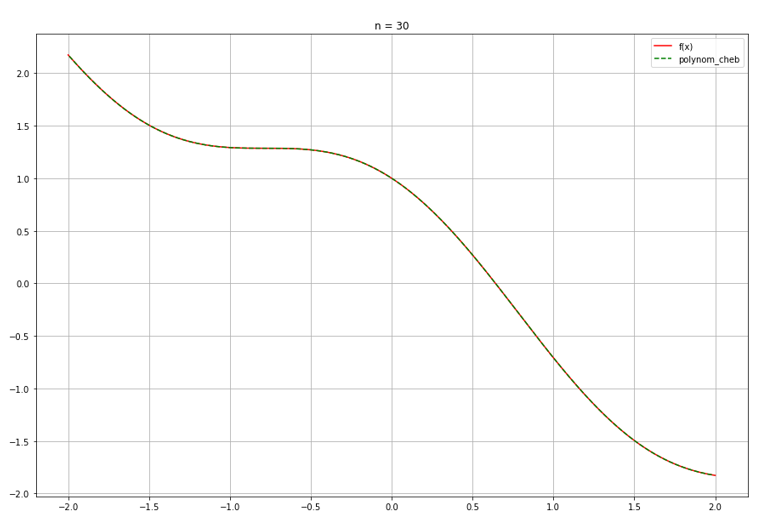
1. **Графики.**

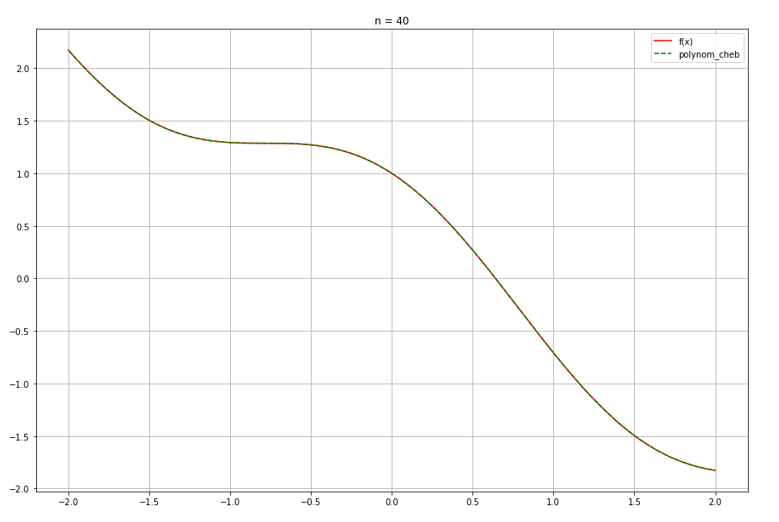
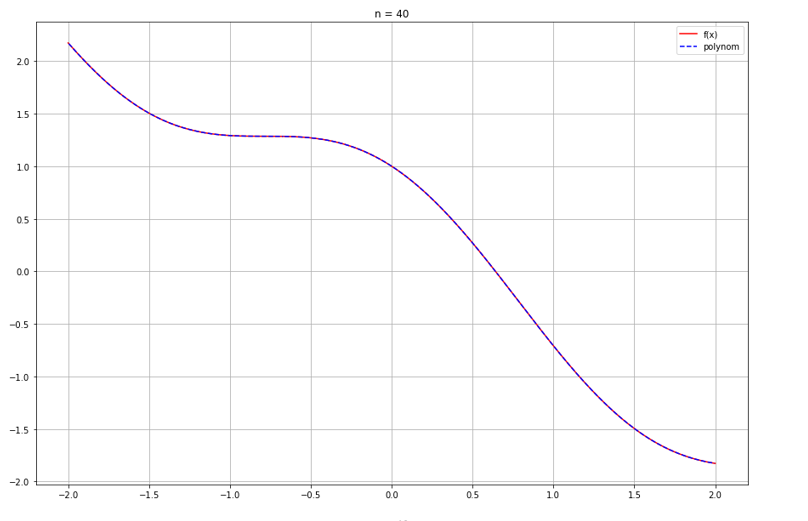


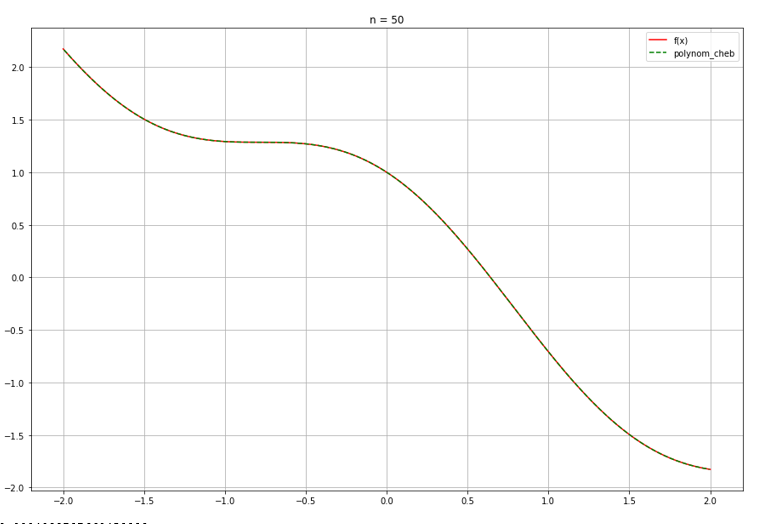
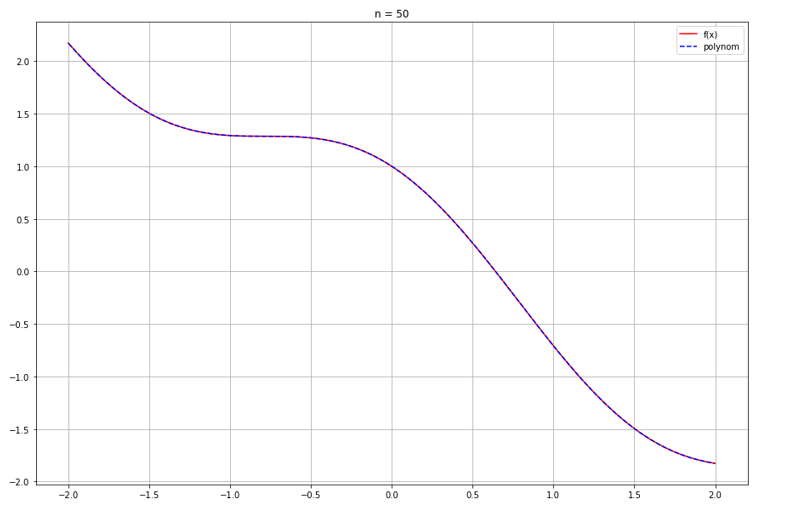


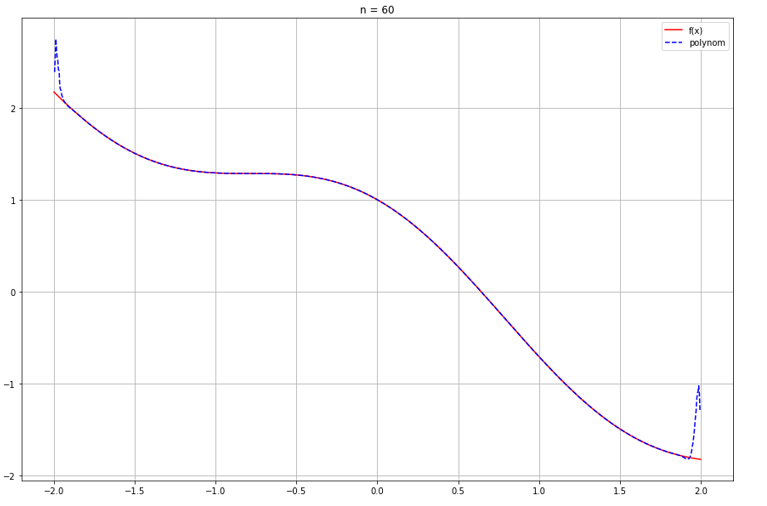


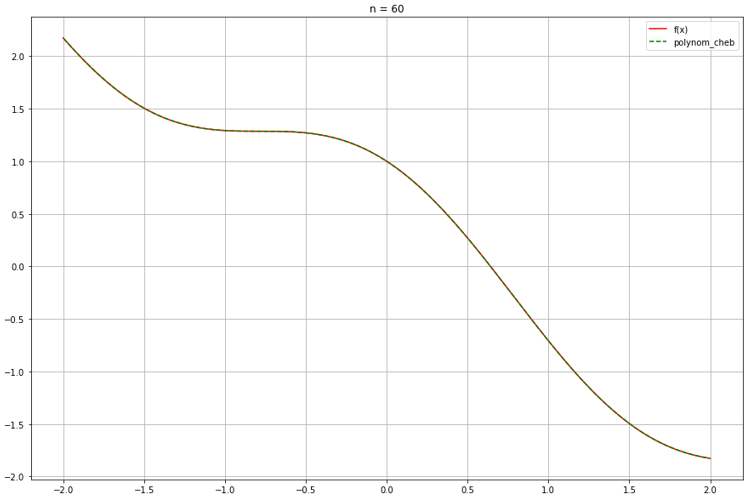


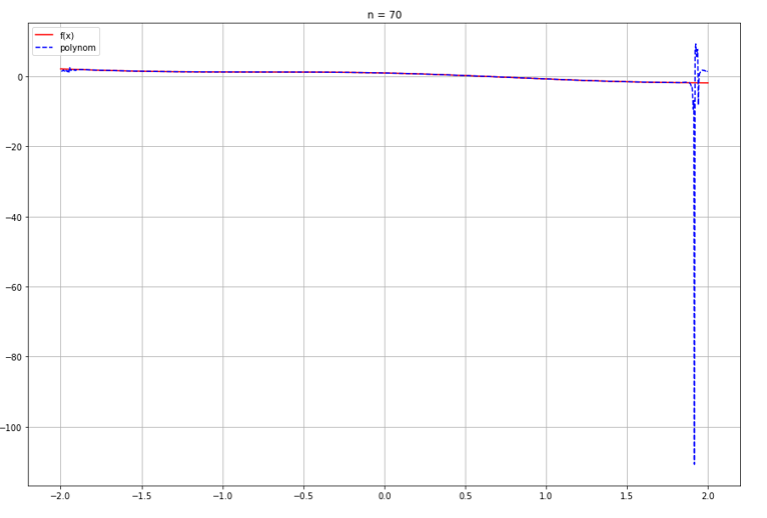


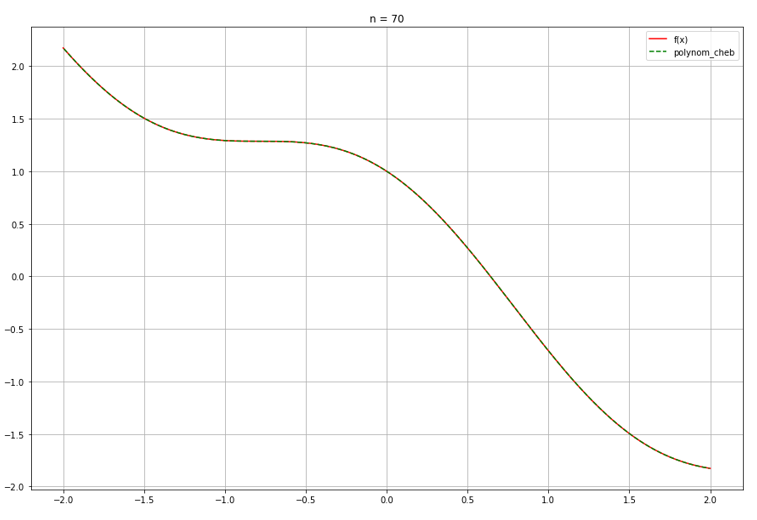


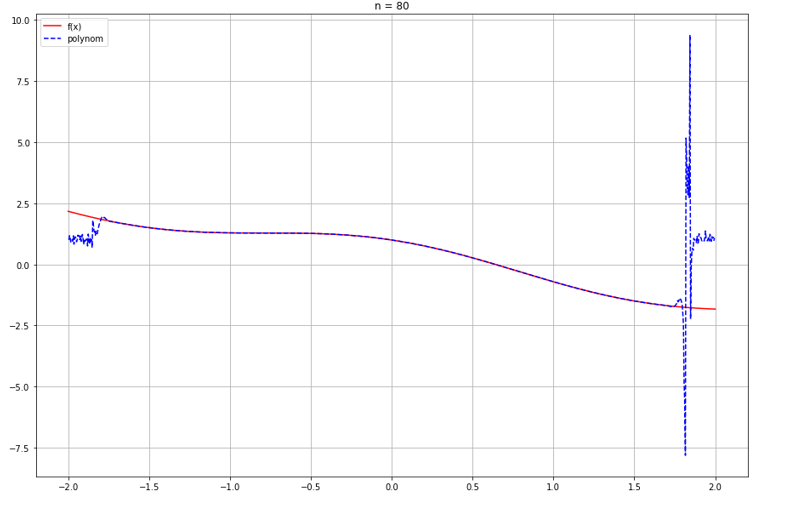


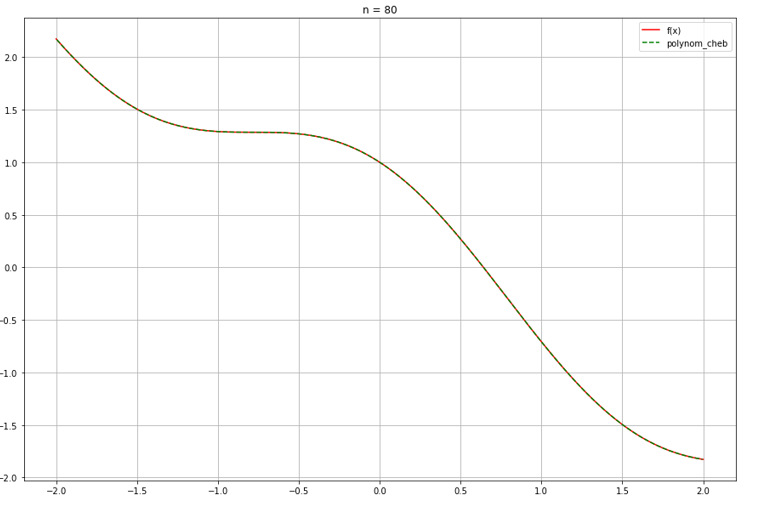


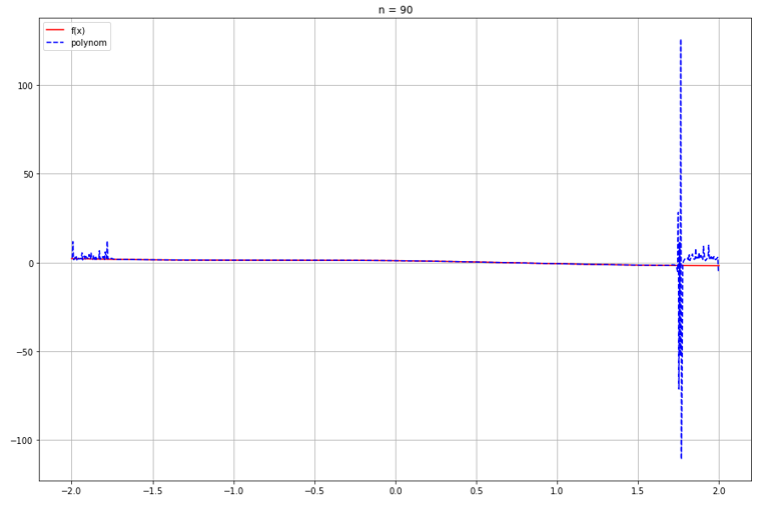


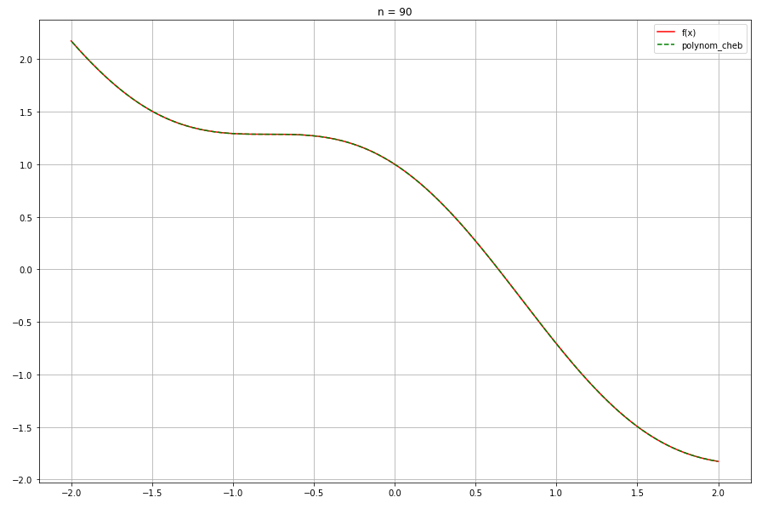


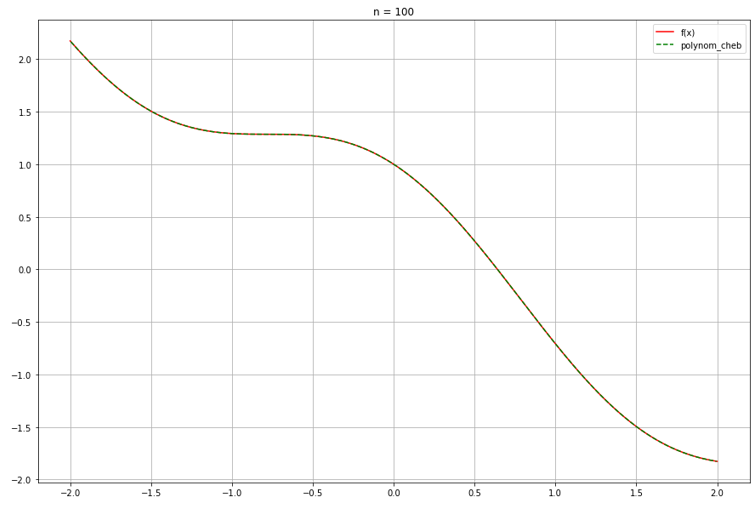
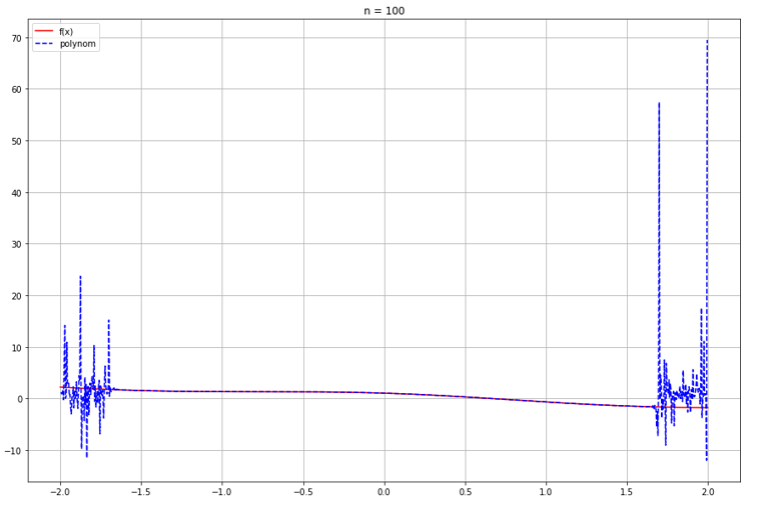












1. **Таблица**

|  |  |  |
| --- | --- | --- |
| N | Норма  (равноотстоящие узлы) | Норма  (чебышевские узлы) |
| 10 | 0.0012723000737486778 | 0.00020785128659173147 |
| 20 | 6.515277206631254e-11 | 3.622657729351886e-13 |
| 30 | 5.714277939716794e-10 | 1.7763568394002505e-15 |
| 40 | 3.7820961362733385e-07 | 2.4424906541753444e-15 |
| 50 | 0.0004098737660453988 | 1.9984014443252818e-15 |
| 60 | 0.7988823946738641 | 1.7763568394002505e-15 |
| 70 | 109.00599986310604 | 2.220446049250313e-15 |
| 80 | 11.161996780941475 | 3.3306690738754696e-15 |
| 90 | 127.70328322270198 | 2.886579864025407e-15 |
| 100 | 71.3247088758354 | 2.886579864025407e-15 |

Список использованной литературы:

1. Б. В. Фалейчик «Методы вычислений» - с.126
2. Б. В. Фалейчик «Методы вычислений» - с.124