«КАЗАНСКИЙ (ПРИВОЛЖСКИЙ) ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

«ИНСТИТУТ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ»

СЕМЕСТРОВАЯ РАБОТА

По дисциплине «Вычислительные методы»

На тему: «Табулирование трансцендентных функций»

Вариант №22

Работу выполнила:
Студентка группы 09-832
Кузьмина В.А.

Работу проверила:
Павлова М.Ф.

Казань, 2021

Оглавление

Постановка задачи

Одна из специальных функций математической физики – интеграл Френеля, определяется следующим образом

$$C(x) = \int_0^x \cos\left(\frac{\pi t^2}{2}\right) dt.$$

Цель задания – изучить и сравнить различные способы приближенного вычисления этой функции.

Ход исследования:

1. Протабулировать C(x) на отрезке [a,b] с шагом h и с точностью ε, основываясь на ряде Тейлора, предварительно вычислив его.

$$C(x) = \sum_{n=0}^{\infty} \frac{(-1)^n * \left(\frac{\pi}{2}\right)^{2n}}{(2n)! * (4n+1)} * x^{4n+1}.$$

Где $a = 0, b = 1.5, h = 0.15, \varepsilon = 10^{-6}$, и получить таким образом таблицу:

x_0	x_1	x_2	…	x_N
f_0	f_1	f_2	…	f_N

где $f_i = C(x_i),\ x_i = a + ih,\ i = \overline{(0, N)}$.

2. По полученной таблице значений найти приближенное значение производной функции C(x), используя следующую формулу.

$$L'(x) = \sum_{k=0}^{N} f_k * l_k(x),$$

$$l_k(x) = \sum_{\substack{j=0 \\ j \neq k}}^{N} \frac{\prod_{i=0}^{N}(x - x_i)}{(x - x_k)(x - x_j)\prod_{\substack{i=0 \\ j \neq k}}^{N}(x_k - x_i)}.$$

В качестве узлов используются:

а) Равномерно распределенные узлы $\{x_i\}_{i=0}^{N}$, где $N = \left(\frac{b-a}{h}\right) + 1 = 11$.

б) Корни полинома Чебышева, находимые по формуле

$$x_i = \frac{a+b}{2} + \frac{b-a}{2} * t_i,$$

где $t_i = \cos\left(\frac{2i+1}{2(N+1)} * \pi\right)$, $\quad N = \left(\frac{b-a}{h}\right) + 1 = 11, \quad i = (\overline{1, N})$.

3. Провести эксперимент:

3.1. Увеличивая число равномерно распределенных узловых точек N, повторять следующие шаги:

- находить приближенное значение производной функции C(x) $= L'(x)$;
- вычислять реальное значение производной C(x):

$$C'(x) = \cos\left(\frac{\pi x^2}{2}\right);$$

- считать максимальную погрешность вычисления приближенного значения производной:

$$\varphi = \max_{\forall x}(|L'(x) - C'(x)|).$$

В итоге, построить график зависимости максимальной погрешности от количества узлов. И найти число узлов, для которого погрешность будет минимальной.

3.2. Повторить эксперимент из пункта 3.1. для узлов, равных корням полинома Чебышева.

4. На той же сетке узлов $\{x_i\}_{i=0}^{n}$ построить таблицу приближенных значений C(x), используя составную квадратурную формулу Гаусса с двумя узлами

$$\int_c^d \varphi(t)dt = \sum_{i=0}^{N} \int_{z_{i-1}}^{z_i} \varphi(t)dt \approx \sum_{i=1}^{N} S_i(\varphi),$$

где $S_i(\varphi) = \frac{h_N}{2}\left[\varphi\left(z_{i-1} + \frac{h_N}{2} * \left(1 - \frac{1}{\sqrt{3}}\right)\right) + \varphi\left(z_{i-1} + \frac{h_N}{2} * \left(1 + \frac{1}{\sqrt{3}}\right)\right)\right].$

z_i – точки разбиения отрезка интегрирования на N частей,

$$z_i = c + i * h_N, \quad h_N = \frac{c-d}{N}.$$

Полученные значения сравнить с ранее полученными значениями функции в узлах разбиения с помощью ряда Тейлора (пункт 1).

5. Провести эксперимент: для максимального узла $x = b$ из отрезка [a, b] найти константу, которая должна прослеживаться при увеличении числа точек разбиения отрезка, по следующей формуле:

$$\frac{|C(x) - S(\varphi)|}{h^4} \approx const.$$

Табулирование функции на отрезке с помощью ряда Тейлора.

В качестве функции будет использоваться интеграл Френеля:

$$C(x) = \int_0^x \cos\left(\frac{\pi t^2}{2}\right) dt.$$

Необходимо протабулировать C(x) на отрезке [a,b] с шагом h и с точностью ε, основываясь на ряде Тейлора, предварительно вычислив его

$$C(x) = \sum_{n=0}^{\infty} \frac{(-1)^n * \left(\frac{\pi}{2}\right)^{2n}}{(2n)! * (4n+1)} * x^{4n+1}.$$

Где $a = 0, b = 1.5, h = 0.15, \varepsilon = 10^{-6}$, и получить таким образом таблицу

x_0	x_1	x_2	…	x_N
f_0	f_1	f_2	…	f_N

$f_i = C(x_i),\ x_i = a + ih,\ i = \overline{(0, N)}.$

Количество узлов: $N = \left(\frac{b-a}{h}\right) + 1 = 11.$

При вычислении ряда Тейлора для оптимизации вычислений можно использовать следующую формулу: $a_{n+1} = a_n * q_n$, где a_i – член ряда.

1. Разложение в ряд Тейлора заданной функции

$$\cos\left(\frac{\pi t^2}{2}\right) = 1 + \frac{0}{1!}x + \frac{0}{2!}x^2 + \frac{0}{3!}x^3 + \frac{-3\pi^2}{4!}x^4 + \cdots + \frac{\pi^4}{384}x^8 + O(x^9)$$

$$= 1 - \frac{\pi^2}{8}x^4 + \frac{\pi^4}{384}x^8 \ldots = \sum_{n=0}^{\infty} \frac{(-1)^n * \left(\frac{\pi}{2}\right)^{2n}}{(2n)!} * x^{4n}$$

2. Вычисление интеграла

$$\int_0^x \cos(\frac{\pi t^2}{2})dt = \int_0^x \sum_{n=0}^{\infty} \frac{(-1)^n * \left(\frac{\pi}{2}\right)^{2n}}{(2n)!} * t^{4n} * dt =$$

$$= \sum_{n=0}^{\infty} \int_0^x \frac{(-1)^n * \left(\frac{\pi}{2}\right)^{2n}}{(2n)!} * t^{4n} * dt = \sum_{n=0}^{\infty} \frac{(-1)^n * \left(\frac{\pi}{2}\right)^{2n}}{(2n)! * (4n+1)} * t^{4n+1} |_0^x =$$

$$= \sum_{n=0}^{\infty} \frac{(-1)^n * \left(\frac{\pi}{2}\right)^{2n}}{(2n)! * (4n+1)} * x^{4n+1} - \sum_{n=0}^{\infty} \frac{(-1)^n * \left(\frac{\pi}{2}\right)^{2n}}{(2n)! * (4n+1)} * 0^{4n+1} =$$

$$= \sum_{n=0}^{\infty} \frac{(-1)^n * \left(\frac{\pi}{2}\right)^{2n}}{(2n)! * (4n+1)} * x^{4n+1}.$$

3. Нахождение q_n для $a_{n+1} = a_n * q_n$

$$a_n = \frac{(-1)^n * \left(\frac{\pi}{2}\right)^{2n}}{(2n)! * (4n+1)} * x^{4n+1},$$

$$a_{n+1} = \frac{(-1)^{n+1} * \left(\frac{\pi}{2}\right)^{2(n+1)}}{\big(2(n+1)\big)! * (4(n+1)+1)} * x^{4(n+1)+1}$$

$$q_n = \frac{a_{n+1}}{a_n} = \frac{\frac{(-1)^{n+1} * \left(\frac{\pi}{2}\right)^{2(n+1)}}{\big(2(n+1)\big)! * (4(n+1)+1)} * x^{4(n+1)+1}}{\frac{(-1)^n * \left(\frac{\pi}{2}\right)^{2n}}{(2n)! * (4n+1)} * x^{4n+1}}$$

$$= \frac{-1 * \left(\frac{\pi}{2}\right)^2 * (4n+1)}{(2n+1) * (2\mathrm{n}+2) * (4n+5)} * x^4.$$

4. Оценка остатка ряда

Пусть знакочередующийся числовой ряд сходится по признаку Лейбница и его сумма равна S.Тогда S_k – частичная сумма ряда, состоящая из k членов. Тогда остаток ряда по модулю должен быть меньше модуля слагаемого, не включенного в S_k.

$|S - S_k| < a_{k+1} < \varepsilon$. Заданная точность $\varepsilon = 10^{-6}$.

Табулирование функции C(x) на отрезке [a,b] с шагом h и точностью ε вызывается в главной функции программы Main. Функция Tailor_1 в качестве входного параметра принимает массив всех узлов x_i, предварительно рассчитанных для заданных границ и размера шага. Результатом работы этой функции является массив рассчитанных функций f_i в соответствующих узлах x_i.

Ниже приведены таблица вычисленных значений функции в узлах и график данной функции.

x_i	f_i
0.00	0.000000000000000
0.15	0.149981264256409
0.30	0.299400976052047
0.45	0.445468228707759
0.60	0.581095446991599
0.75	0.693525990773008
0.90	0.764823019930867
1.05	0.775909458607293
1.20	0.715435968084464
1.35	0.592233877211939
1.50	0.444812266146551


Вычисляемая функция по Тейлору
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
f(x)
0
0.5
1
1.5
x

Поиск приближенного значения производной функции

Необходимо по полученной таблице значений найти приближенное значение производной функции C(x), используя формулу

$$L'(x) = \sum_{k=0}^{N} f_k * l_k(x), \text{где}$$

$$l_k(x) = \sum_{\substack{j=0 \\ j\neq k}}^{N} \frac{\prod_{i=0}^{N}(x - x_i)}{(x - x_k)(x - x_j)\prod_{\substack{i=0 \\ j\neq k}}^{N}(x_k - x_i)}.$$

В качестве узлов будут использоваться:

а) Равномерно распределенные узлы $\{x_i\}_{i=0}^{N}$, где $N = \left(\frac{b-a}{h}\right) + 1 = 11.$

б) Корни полинома Чебышева, находимые по формуле

$$x_i = \frac{a+b}{2} + \frac{b-a}{2} * t_i,$$

где $t_i = \cos\left(\frac{2i+1}{2(N+1)} * \pi\right), \quad N = \left(\frac{b-a}{h}\right) + 1 = 11, \quad i = (\overline{1, N}).$

Для решения поставленной задачи в программе были реализованы следующие функции:

- L_2 – метод для вычисления приближенного значения производной функции по заданной формуле. В качестве входных данных принимает значение конкретного узла, в котором считается производная, количество узлов разбиения, массив всех узлов разбиения и массив соответствующих им функций, рассчитанных по Тейлору на первом шаге.

- X_cheb используется для получения корней полинома Чебышева. Параметры данной функции – границы интервала a, b и количество узлов. Возвращаемое значение – массив точек разбиения.

Вычисленное приближенное значение сравнивалось с реальным значением производной заданной функции:

$$F_i = \cos\left(\frac{\pi x_i^2}{2}\right)$$

Для этого находилась погрешность φ:

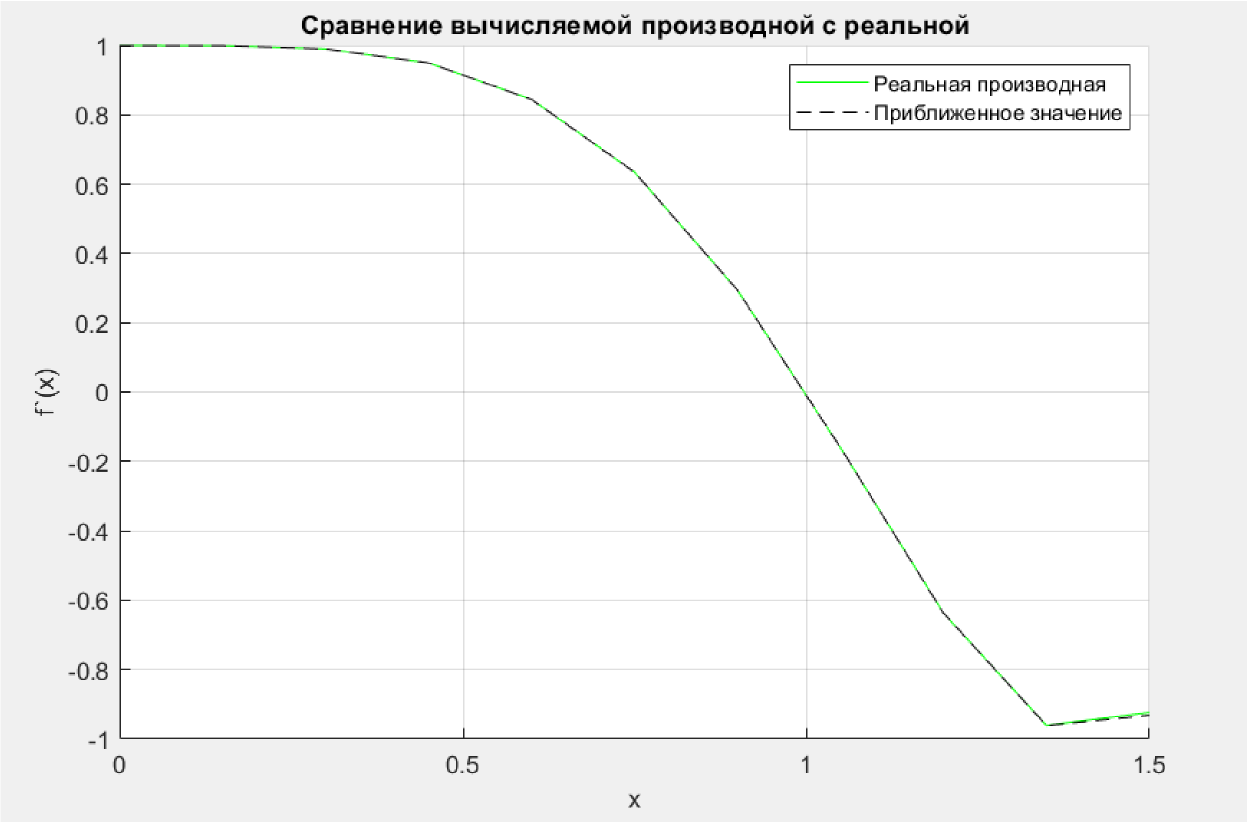
$$\varphi_i = |L(x_i) - F_i|.$$

Ниже приведены результаты вычислений приближенного и реального значений производной заданной функции по сетке равноудаленных узлов.

x_i	F_i	$L(x_i)$	φ_i
0.00	1.000000000000000	0.999647993179737	0.0003520068
0.15	0.999375504106507	0.999416324934808	0.0000408208
0.30	0.990023657716558	0.990013167207284	0.0000104905
0.45	0.949835679280918	0.949840210895305	0.0000045316
0.60	0.844327925502015	0.844324959126875	0.0000029664
0.75	0.634393284163645	0.634396093243414	0.0000028091
0.90	0.294040325232304	0.294036498610696	0.0000038266
1.05	0.160311891908391	-0.160306050678369	0.0000058412
1.20	0.637423989748690	-0.637481555112473	0.0000575654
1.35	0.961381873914186	-0.961887555821553	0.0005056819
1.50	0.923879532511287	-0.932274247830566	0.0083947153

Выводы по полученным данным:

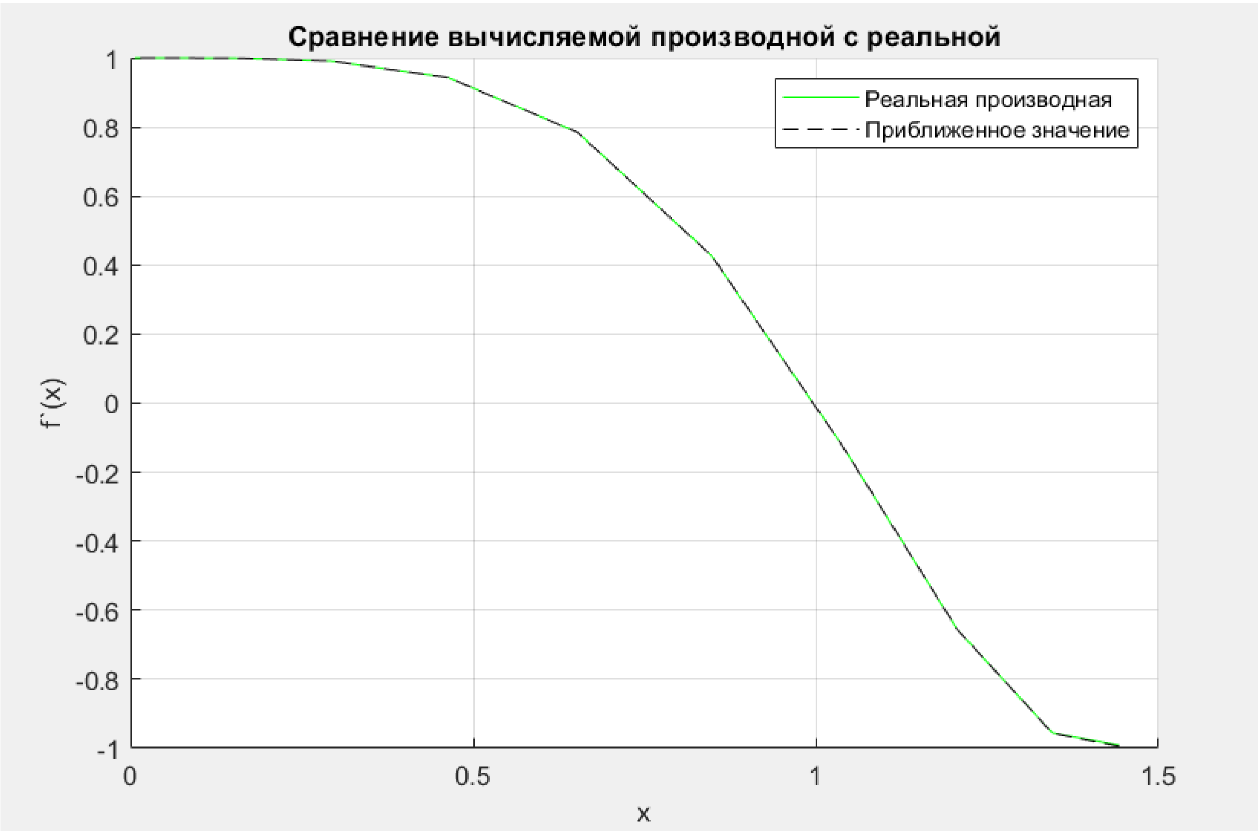
- максимальная погрешность = 0.0083947153, она соответствует последнему узлу разбиения.
- погрешность вычисления приближенного значения больше на концах отрезка, наиболее точные вычисления в центральных узлах разбиения.

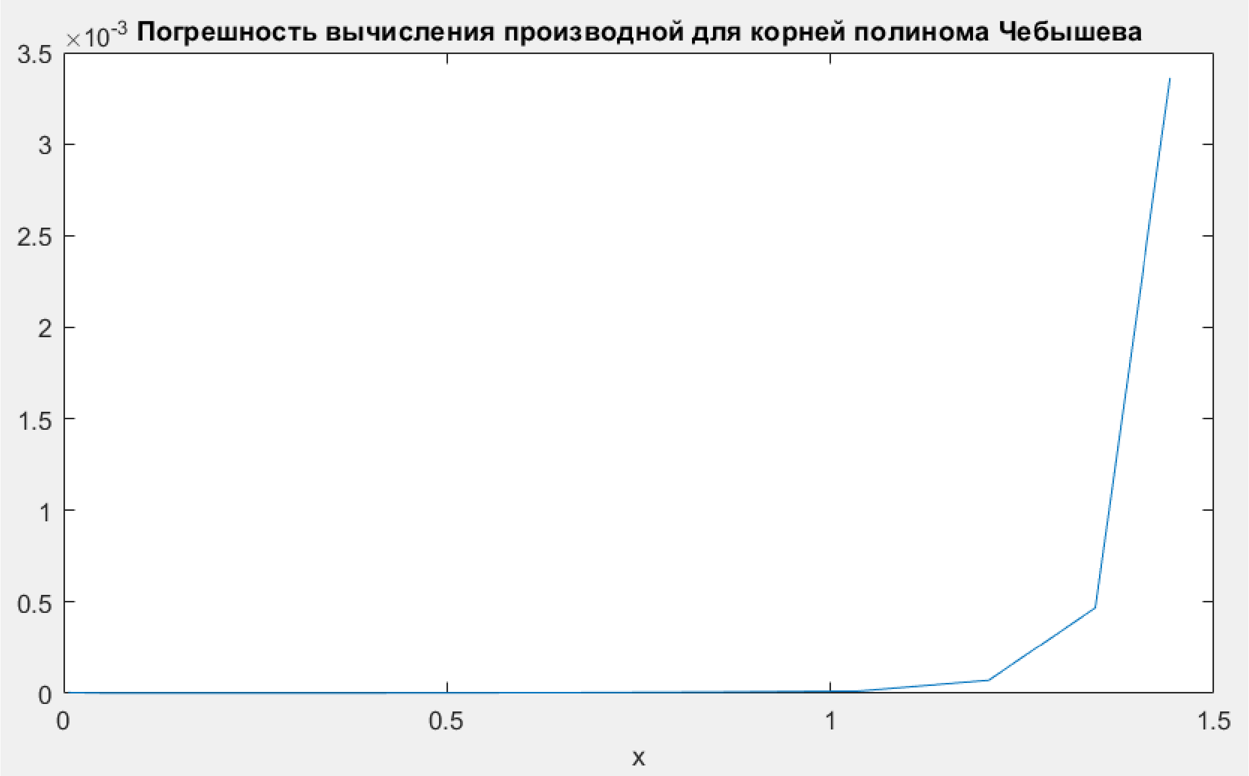

Сравнение вычисляемой производной с реальной
Реальная производная
Приближенное значение
f'(x)
x
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
0.5
1
1.5


Погрешность вычисления производной для равноудаленных узлов
×10^-3
9
8
7
6
5
4
3
2
1
0
0
0.5
1
1.5
x

Результаты аналогичных вычислений для узлов разбиения, равных корням полинома Чебышева.

x_i	F_i	$L(x_i)$	φ_i
1.443	-0.991718429446679	-0.995080998662318	0.0033625692
1.345	-0.955360297720445	-0.955829622466628	0.0004693247
1.207	-0.656365064679369	-0.656436983508515	0.0000719188
1.037	-0.118153662104877	-0.118140847646607	0.0000128145
0.848	0.427306114991492	0.427298796696699	0.0000073183
0.652	0.785082124807806	0.785086375170475	0.0000042504
0.463	0.943846156331416	0.943843424404034	0.0000027319
0.293	0.990868137320397	0.990870119274652	0.0000019820
0.155	0.999288267419293	0.999286558737542	0.0000017087
0.057	0.999986894317255	0.999988870718606	0.0000019764
0.006	0.999999997908957	0.999995148478649	0.0000048494





Выводы по полученным данным:

- максимальная погрешность = 0.0033625692, она соответствует первому корню полинома Чебышева, $x = 1.443$.
- погрешность вычисления приближенного значения больше на начальных корнях полинома Чебышева, то есть на больших x.

Сравнив результаты, полученные при использовании равноудаленных узлов и узлов, соответствующих корням полинома Чебышева, можно заметить, что максимальная погрешность меньше, если проводить вычисления на корнях полинома Чебышева.

Зависимость максимальной погрешности приближенного значения производной функции от количества узлов разбиения

Целью эксперимента является поиск такого числа узлов разбиения, для которого погрешность вычисления приближенного значения производной функции будет минимальной.

Ход эксперимента: увеличивать число узловых точек N и выполнять следующие шаги:

- находить приближенное значение производной функции C(x) $= L'(x)$
- вычислять реальное значение производной C(x):

$$C'(x) = \cos\left(\frac{\pi x^2}{2}\right)$$

- считать максимальную погрешность вычисления приближенного значения производной:

$$\varphi = \max_{\forall x}(|L'(x) - C'(x)|)$$

По полученным данным построить график зависимости максимальной погрешности от количества узлов.

1. Для равномерно распределенных узлов.

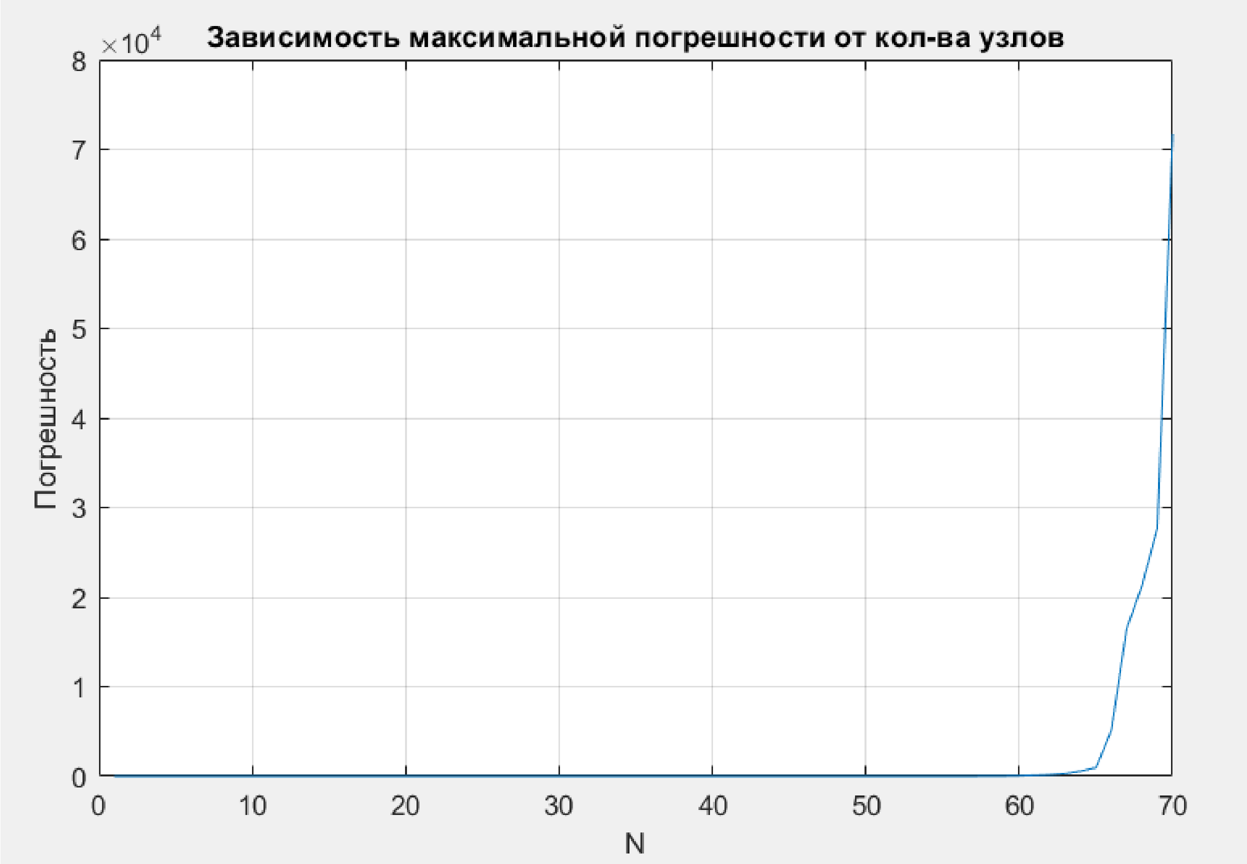
Ниже приведена таблица со списком максимальных погрешностей вычисления, полученных для соответствующего количества узлов N.

N будет изменяться от 1 до 70.

N	φ
1	1.000000000000000
2	1.220421043275654
3	0.552861131296986
4	0.647469771179810

5	0.433600364741580
6	0.118684016207321
7	0.050072819599556
8	0.033612403374392
9	0.008913340859178
10	0.007477374020498
…	
20	0.007413262017805
…	
30	0.007413228375667
…	
40	0.007386420497254
41	0.007223882476480
42	0.007596717913173
43	0.007588274446099
44	0.008654686394116
45	0.009048501835820
46	0.013496434177497
47	0.015768429313190
48	0.004825869127921
49	0.028753115726980
50	0.065547868813263
51	0.222011153339344
52	0.125312368870389
53	0.950679990815579
54	1.126082089601381
55	3.468156396566008
56	3.430454588130110
57	6.194485203795105

58	25.024302644033192
59	28.876183854908042
60	55.389079008272468
61	110.852906708895944
62	165.769653483413862
63	288.675983837338549
64	563.444498613756764
65	954.700768789851281
66	5095.135727509245953
67	16474.151732359543530
68	21263.243579813690303
69	27732.173480335983186
70	71730.453264973344631
min:	
48	0.004825869127921



По данным таблицы можно заметить, что при слишком маленьких $N < 8$ и при слишком больших $N > 53$ погрешность вычисления более высокая. Кроме того, при $N \to +\infty$ погрешность резко растет. Минимальная разница в вычислении, равная 0.0048, наблюдается при количестве узлов = 48.

Ниже приведен график зависимости на наиболее «интересном» интервале.

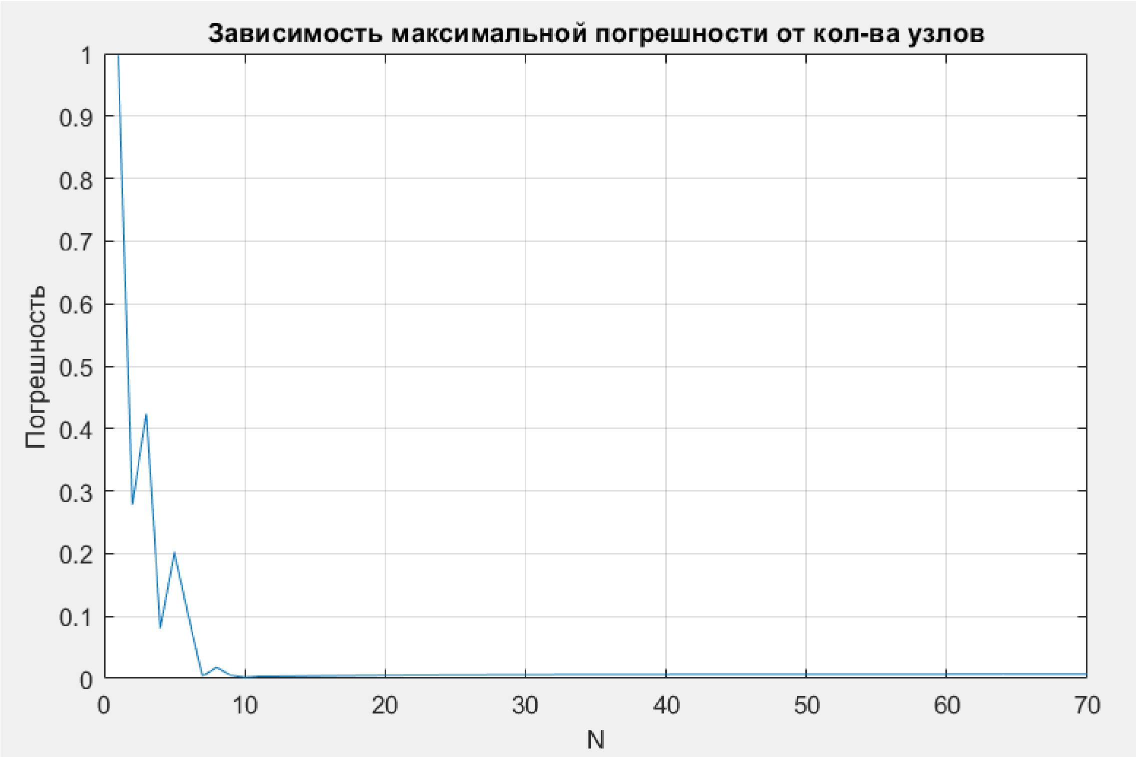
2. Для корней полинома Чебышева.

Ниже приведена таблица со списком максимальных погрешностей вычисления, полученных для соответствующего количества узлов N.

N будет изменяться от 1 до 70.

N	φ
1	0.997128663210201
2	0.278663181692772
3	0.423792666969474
4	0.080784085476559
5	0.202676435889965
6	0.099346247959716
7	0.004276054659033
8	0.018070590942670

9	0.005596474063926
10	0.001803101010045
11	0.003362569215639
12	0.003518985200489
13	0.003756330804871
14	0.004091600525029
15	0.004411444635727
16	0.004688612393400
17	0.004929170289182
18	0.005140702950899
19	0.005328050275166
20	0.005494600442517
…	
30	0.006462428616877
…	
40	0.006854017078065
…	
50	0.007046913160603
…	0.007146750657911
60	0.007155266988177
…	
70	0.007221957597063
min:	
10	0.001803101010045



По данным графика и таблицы можно заметить, что при слишком маленьких $N < 8$ погрешность вычисления более высокая. Кроме того, при $N > 11$ значение погрешности перестает быть неустойчивым и с ростом N тоже постепенно увеличивается. Минимальная разница в вычислении, равная 0.0018, наблюдается при количестве узлов = 10.

Ниже приведен график зависимости на наиболее «интересном» интервале.

Расчет приближенных значений функции с помощью составной квадратурной формулы Гаусса

Необходимо рассчитать приближенные значения C(x) на сетке узлов $\{x_i\}_{i=0}^{n}$ с помощью составной квадратурной формулы Гаусса с двумя узлами.

$$\int_c^d \varphi(t)dt = \sum_{i=0}^{N} \int_{z_{i-1}}^{z_i} \varphi(t)dt \approx \sum_{i=1}^{N} S_i(\varphi),$$

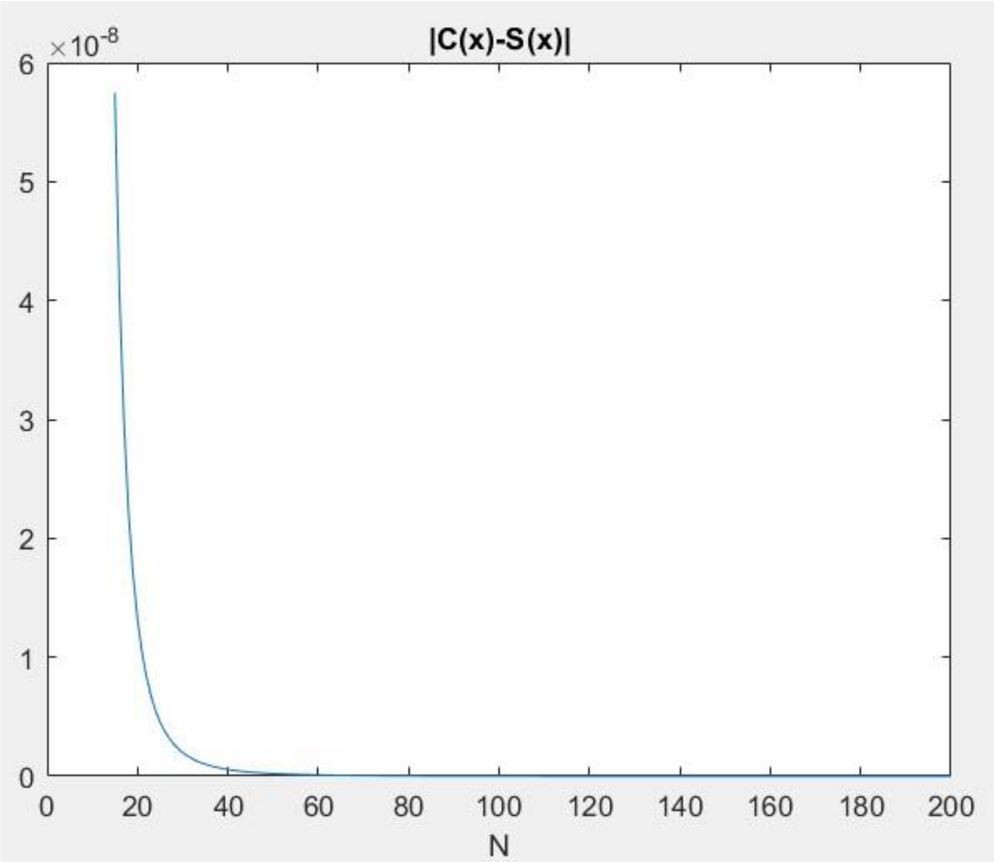
где $S_i(\varphi) = \frac{h_N}{2}\left[\varphi\left(z_{i-1} + \frac{h_N}{2} * \left(1 - \frac{1}{\sqrt{3}}\right)\right) + \varphi\left(z_{i-1} + \frac{h_N}{2} * \left(1 + \frac{1}{\sqrt{3}}\right)\right)\right.$.

z_i – точки разбиения отрезка интегрирования на N частей,

$$z_i = c + i * h_N, \quad h_N = \frac{c - d}{N}.$$

В таблице представлены полученные значения, в сравнении с ранее полученными значениями функции в узлах разбиения с помощью ряда Тейлора (из пункта 1).

x_i	$C(x)$	$S(x)$	$\lvert C(x) - S(x)\rvert$
0.00	0.000000000	0.000000000	0.000000000
0.15	0.149981264	0.149981784	0.000000520
0.30	0.299400976	0.299401996	0.000001020
0.45	0.445468229	0.445469620	0.000001392
0.60	0.581095447	0.581096811	0.000001364
0.75	0.693525991	0.693526497	0.000000506
0.90	0.764823020	0.764821445	0.000001575
1.05	0.775909459	0.775904780	0.000004679
1.20	0.715435968	0.715430129	0.000005839
1.35	0.592233877	0.592259572	0.000025695
1.50	0.444812266	0.445260851	0.000448585

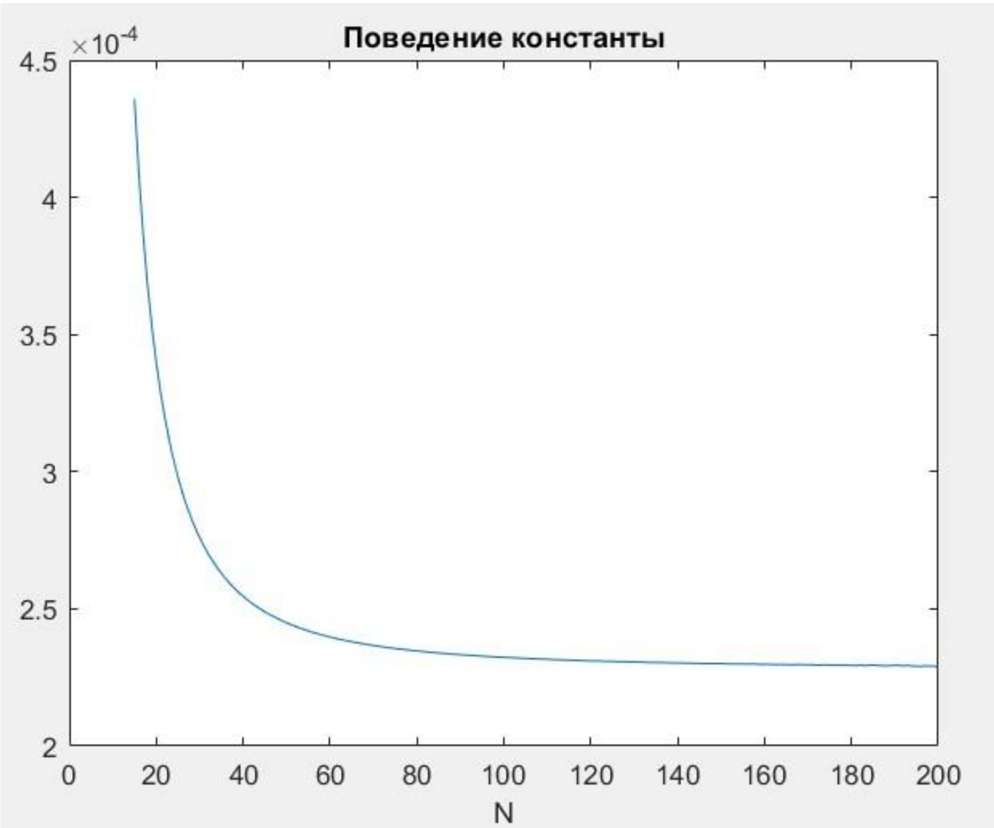

|C(x)-S(x)|
×10-8
6
5
4
3
2
1
0
0
20
40
60
80
100
120
140
160
180
200
N

Поиск константного значения разницы в приближенных вычислениях производной

Проведем следующий эксперимент: для максимального узла $x = b$ из отрезка [a, b] попробуем найти константу, которая должна прослеживаться при увеличении числа точек разбиения отрезка, по следующей формуле.

$$\frac{|C(x) - S(\varphi)|}{h^4} \approx const.$$

В качестве диапазона числа узлов разбиения, на котором будем искать константу, возьмем промежуток $N = [\overline{15,200}]$.



Найденной константе соответствует значение ≈ 0.000229.

Выводы

1. Для вычисления приближенного значения производной функции C(x) с помощью формулы L(x) целесообразно пользоваться не равномерно распределенными узлами, а узлами, соответствующими корням полинома Чебышева, так как погрешность вычислений во втором случае меньше.

2. Погрешность вычисления приближенного значения производной функции на равноудаленных точках будет минимальной при количестве разбиений $\mathrm{N} = 48$. При $N > 53$ погрешность вычисления становится нестабильной и резко растет вверх.

3. Погрешность вычисления приближенного значения производной функции на корнях полинома Чебышева будет минимальной при количестве узлов N = 10. При $N > 11$ значение погрешности относительно стабильно и постепенно увеличивается с ростом N.

4. Модуль разницы в рассчитанных приближенных значениях C(x) с помощью составной квадратурной формулы Гаусса с двумя узлами и с помощью ряда Тейлора на сетке узлов $\{x_i\}_{i=0}^{n}$ не превышает 0.000448585.

5. При $N > 180$ погрешность рассчитанных приближенных значений C(x) в точке $x = b$ с помощью составной квадратурной формулы Гаусса с двумя узлами и с помощью ряда Тейлора, деленная на значение шага в четвертной степени, становится статичной величиной, равной 0.000229.

Листинг программы

Основной модуль

```
a = 0;
b = 1.5;
kolvo_node = 11;
h = (b - a) / (kolvo_node - 1);
format long
x_i_arr = a:h:b;  % равноудаленная сетка узлов
x_i_arr = X_cheb(a, b, kolvo_node);  % узлы Чебышева

        % задание 1
        % рассчет f_i (=C^i(x))
f_i_arr = Tailor_1(x_i_arr);
for i = 1:1:size(x_i_arr, 2)
     fprintf('x_i = %0.4f, f_i = %0.15f\n', x_i_arr(i), f_i_arr(i));
end
plot(x_i_arr, f_i_arr)
grid on
xlabel('x')
ylabel('f(x)')
title('Вычисляемая функция по Тейлору')

        % задание 2
        % рассчет L'(x)
L_list = [];
count = size(f_i_arr, 2); % = n
for x_i = x_i_arr
    L_list(end+1) = L_2(x_i, count, x_i_arr, f_i_arr);
    fprintf('x_i = %0.2f, L_i = %0.15f\n', x_i, L_list(end));
end
```

```
        % сравнение с реальной производной - график
diff_l = cos((pi*x_i_arr.^2)/2);
figure
hold on
plot(x_i_arr, diff_l, 'g')
plot(x_i_arr, L_list, 'k--')
grid on
xlabel('x')
ylabel('f`(x)')
legend('Реальная производная', 'Приближенное значение')
title('Сравнение вычисляемой производной с реальной')
hold off

        % погрешность при таких n и h
Accuracy = L_list - diff_l;
figure
plot(x_i_arr, abs(Accuracy))
xlabel('x')
title('Погрешность вычисления производной')
fprintf('Максимальная по модулю погрешность = %0.10f\n',
max(abs(Accuracy)));

        % задание 3
        % Рассчет квадратур. ф-лы Гаусса
g_i_arr = [];
for x_i=x_i_arr
    g_i_arr(end+1) = Gauss_3(a, x_i, h);
end
fprintf('%0.9f\n', g_i_arr);

        % Поиск константы для h^4
eps = 10^(-25);
accur_Gauss = [];
```

```
const_list = [];
N_arr = 15:1:200; % N варьируем
for N = N_arr
    new_h = (b - a) / (N - 1);
    Tailor_F = Tailor_3(a:new_h:b, eps);
    Gauss_S = Gauss_3(a, b, new_h);
    accur = abs(Tailor_F - Gauss_S);
    accur_Gauss(end+1) = accur;
    const_list(end+1) = accur / (new_h^4);
end
fprintf('%0.10f\n', const_list);
figure
plot(N_arr, accur_Gauss)
xlabel('N')
% title('|C(x)-S(x)|')
title('Разница приближенных значений по Тейлору и Гауссу')
figure
plot(N_arr, const_list)
xlabel('N')
title('Поведение константы')
```

Функция для расчета корней полинома Чебышева

```
function cheb_i_arr = X_cheb(a, b, n)
cheb_i_arr = [];
for i = 1:1:n
    x_i = (a+b)/2 + ((b-a)/2) * cos(((2*i + 1)*pi)/(2*(n + 1)));
    cheb_i_arr(end+1) = x_i;
end
```

Функция для расчета ряда Тейлора

```
function f_i_arr = Tailor_1(x_i_arr)
```

```
f_i_arr = [];
for x_i = x_i_arr
    f_i = 0; % = C_i(x)
    n = 0;
    eps = 10^(-6);
    koef = (((-1)^n) * (pi/2)^(2*n)) / (factorial(2*n) * (4*n + 1));
    while (abs(koef) > eps)
        f_i = f_i + (koef * x_i^(4*n + 1));
        q = ((-1) * ((pi/2)^2) * (4*n + 1)) / ((2*n + 1) * (2*n + 2) *
(4*n + 5));
        koef = koef * q; % считаю заранее след
        n = n + 1;
    end
    f_i_arr(end+1) = f_i;
end
```

Функция для расчета L(x)

```
function L = L_2(x, count, x_i_arr, f_i_arr)
L = 0;
for k = 1:1:count
    l_k = 0;
    for j = 1:1:count
        if j == k
            continue;  % пропуск шага
        end
        numer = 1;
        denomin = 1;
        for i = 1:1:count
            if i == k
                continue;  % пропуск шага, чтобы избежать деления на 0
            end
            denomin = denomin * (x_i_arr(k) - x_i_arr(i));
```

```
            if i == j
                continue;  % пропуск шага, чтобы избежать деления на 0
            end
            numer = numer * (x - x_i_arr(i));
        end

        koef = numer / denomin;
% числитель / знаменатель ((X - x_i_arr(k)) * (X - x_i_arr(j)))
        l_k = l_k + koef;
    end
    L = L + (f_i_arr(k) * l_k);
End
```

Модуль для точности приближенного вычисления производной

```
a = 0;
b = 1.5;
start_node = 1;
kolvo_node = 14;
format long
Max_accur_arr = [];

for n = start_node:1:kolvo_node
    h = (b - a) / (n - 1);
%     x_i_arr = a:h:b;  % равноудаленная сетка узлов
    x_i_arr = X_cheb(a, b, n);  % узлы Чебышева

            % рассчет f_i (=C^i(x))
    f_i_arr = Tailor_1(x_i_arr);

            % рассчет L'(x)
    L_list = [];
```

```
    count = size(f_i_arr, 2); % = n
    for x_i = x_i_arr
        L_list(end+1) = L_2(x_i, count, x_i_arr, f_i_arr);
    end
            % погрешность;
    diff_list = cos((pi*x_i_arr.^2)/2);
    Accuracy = L_list - diff_list;
    Max_accur_arr(end+1) = max(abs(Accuracy));
%     fprintf('%0.0f, %0.15f\n', n, Max_accur_arr(end));
end


[Min_accur, min_acc_nodes] = min(Max_accur_arr);
min_acc_nodes = min_acc_nodes + start_node - 1; % если начинали не с
кол-во узлов = 1
fprintf('Минимальная погрешность (из максимальных) = %0.10f, \nСоответ
кол-во узлов = %0.1f\n', Min_accur, min_acc_nodes);


        % погрешность - график
plot(start_node:1:kolvo_node, Max_accur_arr)
grid on
xlabel('N')
ylabel('Погрешность')
title('Зависимость максимальной погрешности от кол-ва узлов')
```

Функция вычисления ряда Тейлора с заданной точностью

```
function f_b = Tailor_3(x_i_arr, eps)


f_i_arr = [];
for x_i = x_i_arr
    f_i = 0; % = C_i(x)
    n = 0;
    koef = (((-1)^n) * (pi/2)^(2*n)) / (factorial(2*n) * (4*n + 1));
```

```
    while (abs(koef) > eps)
        f_i = f_i + (koef * x_i^(4*n + 1));
        q = ((-1) * ((pi/2)^2) * (4*n + 1)) / ((2*n + 1) * (2*n + 2) *
(4*n + 5));
        koef = koef * q; % считаю заранее след
        n = n + 1;
    end
    f_i_arr(end+1) = f_i;
end
f_b = f_i_arr(end);
```

Функция для расчетов с помощью составной квадратурной формулы Гаусса

```
function S_i_sum = Gauss_3(c, d, h)
z_i_arr = c:h:(d-h);
S_i_arr = [];

for z_i = z_i_arr
    z1 = z_i + (h/2) * (1 - (1/sqrt(3)));
    z2 = z_i + (h/2) * (1 + (1/sqrt(3)));
    S_i = (h/2) * (cos((pi*z1^2)/2) +  (cos((pi*z2^2)/2)));
    S_i_arr(end+1) = S_i;
end
S_i_sum = sum(S_i_arr);
```