**Токаева Александра, 409 группа**

**Отчет по практикуму на ЭВМ**

**Задание 1.5**: загадана некоторая функция f; дан отрезок [a, b],

даны n точек на этом отрезке и даны значения

загаданной функции в этих точках. Нужно по этим данным построить для загаданной функции на данном отрезке интерполяционный многочлен степени :

.

Проблема заключается в том, что нам дана 1 лишняя точка, ведь многочлен степени (а именно все его штук коэффициентов) однозначно строится по точке, а у нас дано точек (то есть неизвестных и уравнений).

Однако есть теорема, согласно которой искомый многочлен будет удовлетворять условию , где это какая-то неизвестная константа. Добавляя это неизвестное к нашим неизвестным , получим неизвестных и уравнений вида

. Найдем коэффициенты как решение этой СЛУ, используя какой-нибудь точный метод решения системы (я буду использовать метод отражений).

Требуется построить этот многочлен по равноотстоящим узлам (то есть ) и по системе чебышевских узлов (то есть ).

Построить графики функции и ее многочлена, а также график зависимости от для своей функции, функции Рунге и функции модуль.

**Решение:** Для запуска программы в текущей директории должен лежать makefile и файл mnog\_lagranzh\_N.cpp.

Запускаем в командной строке make, а потом строки наподобие ./a.out -1 1 10 ravnom myfunc

Инструкция по использованию:

Usage: ./a.out a b n ravnom/chebushov myfunc/runge/modul

Здесь a,b – концы отрезка, n-число узлов ( в случае равноотстоящих узлов оба конца отрезка являются первым и последним узлом соответственно, в случае чебышевских узлов – все n узлов лежат внутри отрезка, концы отрезка не являются узлами), ravnom/chebushov отвечает за вид узлов; вид загаданной функции регулируется функцией double func(double x).

Функция int generate\_input(double a, double b, int n, char\* tip\_uzlov, const char\* filename) записывает входные данные (узлы и значения функции func) в файл filename, потом функция main открывает этот файл, считывает массив точек в mas1, массив значений в mas2, вычисляет значение в исходных узлах и в добавочных узлах (по два добавочных узла между каждыми старыми узлами) и записывает их в файл типа output\_5ravnomrunge.txt в 4 столбика в формате

x\_i, f(x\_i), P\_n\_2(x\_i), f(x\_i)- P\_n\_1(x\_i).

Еще дополнительно в файл commands\_for\_gnuplot.txt дописывается что-то наподобие

set terminal png size 640,480

set output 'output\_10ravnommyfunc.txt.png'

plot 'output\_10ravnommyfunc.txt' using 1:2 with lines, 'output\_10ravnommyfunc.txt' using 1:3 with lines, 'output\_10ravnommyfunc.txt' using 1:2 with points

А в файл ‘file\_of\_h.txt’ записываются значения n h в 2 столбика.

После того, как все нужные расчеты были выполнены, следует запустить в командной строке gnuplot, после приглашения написать

>gnuplot load ‘comands\_for\_gnuplot.txt’

И все графики сами нарисуются и сохранятся в файлы типа

'output\_10ravnommyfunc.txt.png'

Наконец, запустив

>gnuplot set terminal png size 640,480

>gnuplot set output 'h\_for\_ravnom\_myfunc.png'

>gnuplot plot ‘file\_of\_h.txt’ with lines

Получим график зависимости h(n).

**Моя функция**

**a)Равномерные узлы**

Запускаем make clean, make, ./a.out -1 1 7 ravnom myfunc

Получаем вывод

Hello!

a=-1.000000 b=1.000000 n=7 is\_ravnom=0 myfunc

ravnom uzl

-1.000000 -0.666667 -0.333333 0.000000 0.333333 0.666667 1.000000

1.753902 -0.498347 -0.465348 -0.416147 -0.130493 1.312999 0.010008

flag=0

**h=-5.841649e-02**

tmp\_str=7

output\_name=output\_7ravnommyfunc.txt

-1.000000 1.753902 1.812318 **-5.841624e-02**

-0.888889 1.506971 0.322552 1.184419e+00

-0.777778 0.252803 -0.350951 6.037547e-01

-0.666667 -0.498347 -0.556764 **5.841639e-02**

-0.555556 -0.651229 -0.540980 -1.102490e-01

-0.444444 -0.565611 -0.462271 -1.033402e-01

-0.333333 -0.465348 -0.406932 **-5.841654e-02**

-0.222222 -0.416006 -0.403936 -1.206991e-02

-0.111111 -0.410028 -0.439986 2.995763e-02

0.000000 -0.416147 -0.474563 **5.841666e-02**

0.111111 -0.397555 -0.454984 5.742898e-02

0.222222 -0.316270 -0.331444 1.517381e-02

0.333333 -0.130493 -0.072076 **-5.841738e-02**

0.444444 0.207285 0.322003 -1.147187e-01

0.555556 0.723817 0.801637 -7.781990e-02

0.666667 1.312999 1.254581 **5.841743e-02**

0.777778 1.543438 1.490452 5.298565e-02

0.888889 0.849226 1.225673 -3.764474e-01

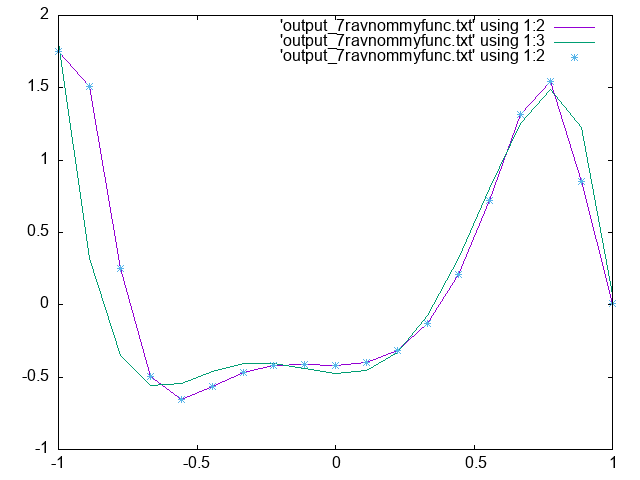
1.000000 0.010008 0.068424 **-5.841699e-02**

Goodbuy!

Видим, что **h=-5.841649e-02**

Причем в 4 столбике в исходных узлах значение разности принимает значения h,-h,h,-h,h,-h, как и предполагалось.

Посмотрим на графики исходной функции и полученного многочлена



Посмотрим график зависимости h от n (для этого нужно последовательно запустить в терминале

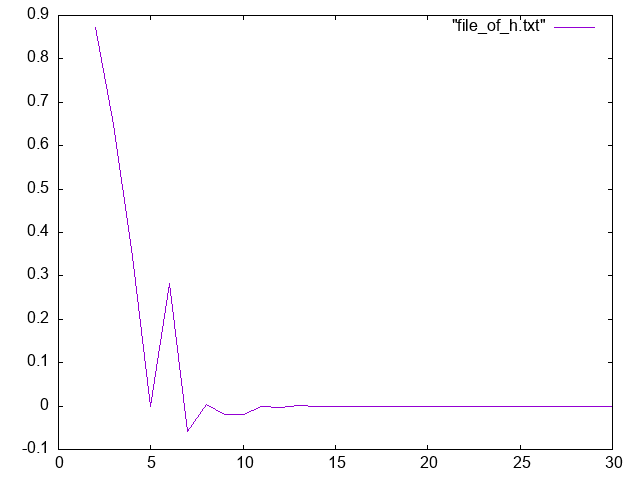
./a.out -1 1 2 ravnom myfunc

./a.out -1 1 3 ravnom myfunc

…………..

./a.out -1 1 30 ravnom myfunc

И посторить график из файла ‘file\_of\_h.txt’

****

Видим, что при росте n число h уменьшается.

**б)Чебышевские узлы**

Запускаем make clean, make, ./a.out -1 1 7 chebushov myfunc

Получаем вывод

Hello!

a=-1.000000 b=1.000000 n=7 is\_ravnom=-15 myfunc

chebushov uzl

flag=0

**h=-1.416061e-01**

tmp\_str=7

output\_name=output\_7chebushovmyfunc.txt

0.974928 0.076922 0.218529 **-1.416068e-01**

0.910562 0.626489 0.903858 -2.773695e-01

0.846197 1.231219 1.269431 -3.821209e-02

0.781831 1.536355 1.394748 **1.416075e-01**

0.665849 1.309197 1.223175 8.602188e-02

0.549866 0.693672 0.792007 -9.833442e-02

0.433884 0.167458 0.309063 **-1.416051e-01**

0.289256 -0.219670 -0.183605 -3.606495e-02

0.144628 -0.381428 -0.468429 8.700016e-02

0.000000 -0.416147 -0.557753 **1.416063e-01**

-0.144628 -0.408936 -0.526290 1.173548e-01

-0.289256 -0.439199 -0.461368 2.216910e-02

-0.433884 -0.554789 -0.413183 **-1.416066e-01**

-0.549866 -0.649882 -0.366044 -2.838378e-01

-0.665849 -0.501239 -0.228571 -2.726679e-01

-0.781831 0.293961 0.152360 **1.416008e-01**

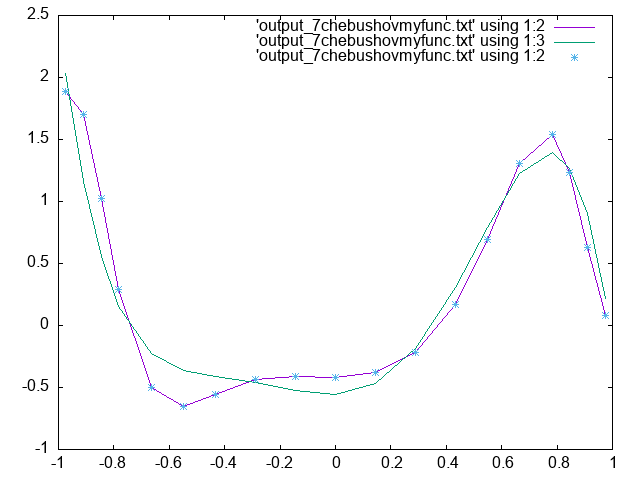
-0.846197 1.027958 0.553827 4.741309e-01

-0.910562 1.702677 1.158187 5.444897e-01

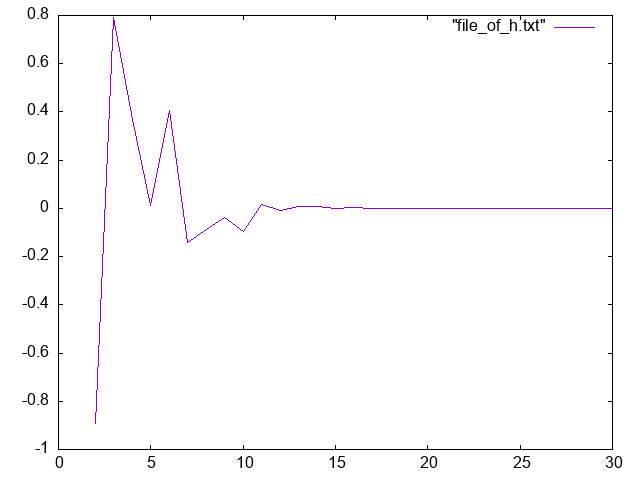
-0.974928 1.889828 2.031434 **-1.416059e-01**

Goodbuy!

Посмотрим на графики исходной функции и полученного многочлена



Посмотрим график зависимости h от n



**Функция Рунге**

**a)Равномерные узлы**

Запускаем make clean, make, ./a.out -1 1 7 ravnom runge

Hello!

a=-1.000000 b=1.000000 n=7 is\_ravnom=0 runge

ravnom uzl

-1.000000 -0.666667 -0.333333 0.000000 0.333333 0.666667 1.000000

0.038462 0.082569 0.264706 1.000000 0.264706 0.082569 0.038462

flag=0

**h=-2.026992e-01**

tmp\_str=7

output\_name=output\_7ravnomrunge.txt

-1.000000 0.038462 0.241161 **-2.026996e-01**

-0.888889 0.048186 -0.092266 1.404520e-01

-0.777778 0.062021 -0.187872 2.498938e-01

-0.666667 0.082569 -0.120130 **2.026986e-01**

-0.555556 0.114731 0.046418 6.831290e-02

-0.444444 0.168399 0.257157 -8.875820e-02

-0.333333 0.264706 0.467405 **-2.026987e-01**

-0.222222 0.447514 0.642405 -1.948917e-01

-0.111111 0.764151 0.757336 6.815159e-03

0.000000 1.000000 0.797301 **2.026992e-01**

0.111111 0.764151 0.757336 6.815159e-03

0.222222 0.447514 0.642405 -1.948917e-01

0.333333 0.264706 0.467405 **-2.026987e-01**

0.444444 0.168399 0.257157 -8.875820e-02

0.555556 0.114731 0.046418 6.831290e-02

0.666667 0.082569 -0.120130 **2.026986e-01**

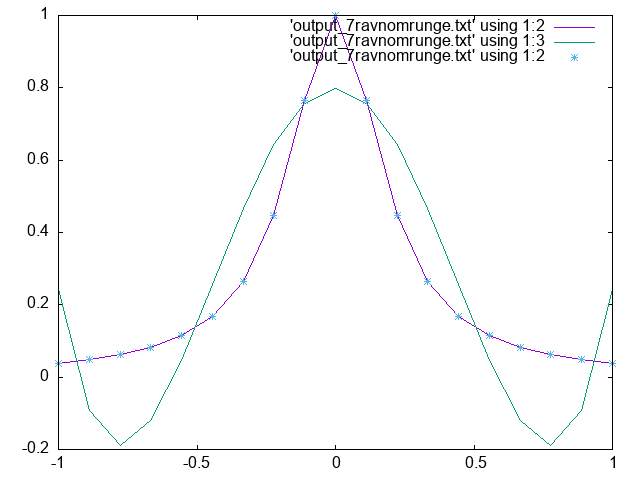
0.777778 0.062021 -0.187872 2.498938e-01

0.888889 0.048186 -0.092266 1.404520e-01

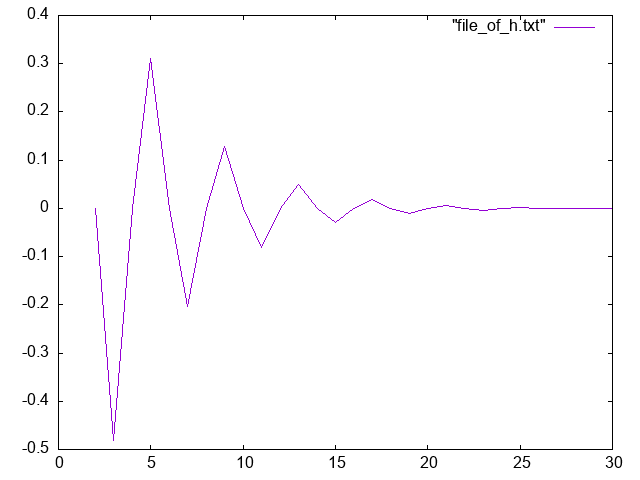
1.000000 0.038462 0.241161 **-2.026996e-01**

Goodbuy!

Посмотрим на графики исходной функции и полученного многочлена



Посмотрим график зависимости h от n



**б)Чебышевские узлы**

Запускаем make clean, make, ./a.out -1 1 7 chebushov runge

Получаем вывод

Hello!

a=-1.000000 b=1.000000 n=7 is\_ravnom=-15 runge

chebushov uzl

flag=0

**h=-1.652970e-01**

tmp\_str=7

output\_name=output\_7chebushovrunge.txt

0.974928 0.040384 0.205681 **-1.652967e-01**

0.910562 0.046023 0.026835 1.918831e-02

0.846197 0.052907 -0.071525 1.244317e-01

0.781831 0.061419 -0.103878 **1.652974e-01**

0.665849 0.082755 -0.037791 1.205458e-01

0.549866 0.116838 0.129889 -1.305079e-02

0.433884 0.175242 0.340540 **-1.652977e-01**

0.289256 0.323444 0.592537 -2.690932e-01

0.144628 0.656628 0.770781 -1.141525e-01

0.000000 1.000000 0.834703 **1.652970e-01**

-0.144628 0.656628 0.770781 -1.141525e-01

-0.289256 0.323444 0.592537 -2.690932e-01

-0.433884 0.175242 0.340540 **-1.652977e-01**

-0.549866 0.116838 0.129889 -1.305079e-02

-0.665849 0.082755 -0.037791 1.205458e-01

-0.781831 0.061419 -0.103878 **1.652974e-01**

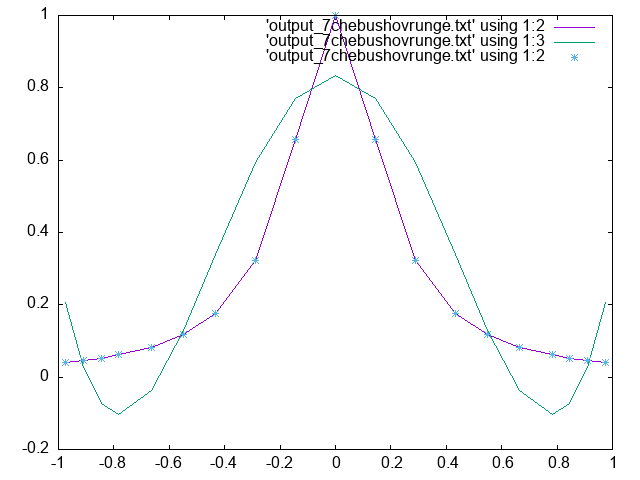
-0.846197 0.052907 -0.071525 1.244317e-01

-0.910562 0.046023 0.026835 1.918831e-02

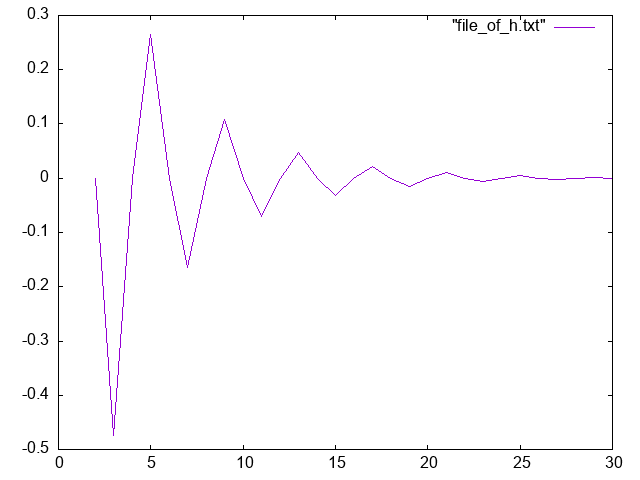
-0.974928 0.040384 0.205681 **-1.652967e-01**

Goodbuy!

Посмотрим на графики исходной функции и полученного многочлена



Посмотрим график зависимости h от n



**Функция модуль**

**a)Равномерные узлы**

Запускаем make clean, make, ./a.out -1 1 7 ravnom modul

Hello!

a=-1.000000 b=1.000000 n=7 is\_ravnom=0 modul

ravnom uzl

-1.000000 -0.666667 -0.333333 0.000000 0.333333 0.666667 1.000000

1.000000 0.666667 0.333333 -0.000000 0.333333 0.666667 1.000000

flag=0

**h=6.250005e-02**

tmp\_str=7

output\_name=output\_7ravnommodul.txt

-1.000000 1.000000 0.937500 **6.250005e-02**

-0.888889 0.888889 0.940415 -5.152599e-02

-0.777778 0.777778 0.860682 -8.290459e-02

-0.666667 0.666667 0.729167 **-6.249994e-02**

-0.555556 0.555556 0.572617 -1.706100e-02

-0.444444 0.444444 0.413666 3.077848e-02

-0.333333 0.333333 0.270833 **6.249999e-02**

-0.222222 0.222222 0.158522 6.370025e-02

-0.111111 0.111111 0.087020 2.409118e-02

0.000000 -0.000000 0.062500 **-6.250005e-02**

0.111111 0.111111 0.087020 2.409118e-02

0.222222 0.222222 0.158522 6.370025e-02

0.333333 0.333333 0.270833 **6.249999e-02**

0.444444 0.444444 0.413666 3.077848e-02

0.555556 0.555556 0.572617 -1.706100e-02

0.666667 0.666667 0.729167 **-6.249994e-02**

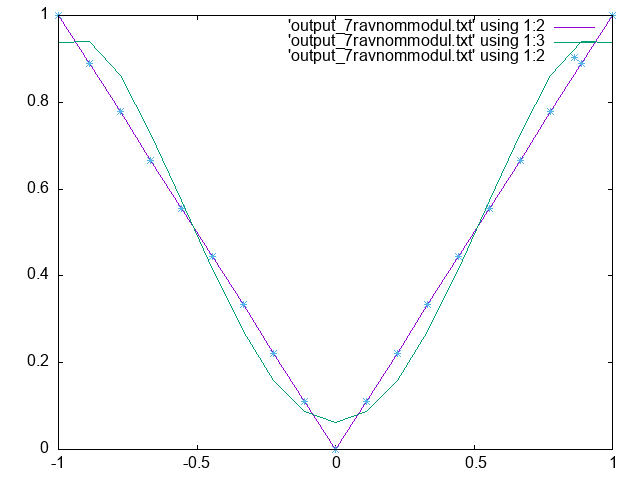
0.777778 0.777778 0.860682 -8.290459e-02

0.888889 0.888889 0.940415 -5.152599e-02

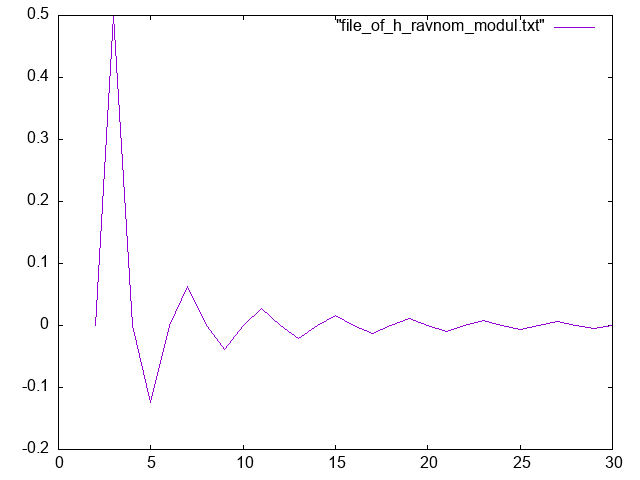
1.000000 1.000000 0.937500 **6.250005e-02**

Goodbuy!

Посмотрим на графики исходной функции и полученного многочлена



Посмотрим график зависимости h от n



**б)Чебышевские узлы**

Запускаем make clean, make, ./a.out -1 1 7 chebushov modul

Получаем вывод

Hello!

a=-1.000000 b=1.000000 n=7 is\_ravnom=-15 modul

chebushov uzl

flag=0

**h=5.358020e-02**

tmp\_str=7

output\_name=output\_7chebushovmodul.txt

0.974928 0.974928 0.921348 **5.358020e-02**

0.910562 0.910562 0.919136 -8.573788e-03

0.846197 0.846197 0.888468 -4.227175e-02

0.781831 0.781831 0.835411 **-5.358020e-02**

0.665849 0.665849 0.700986 -3.513718e-02

0.549866 0.549866 0.541096 8.770416e-03

0.433884 0.433884 0.380304 **5.358020e-02**

0.289256 0.289256 0.208234 8.102205e-02

0.144628 0.144628 0.093660 5.096788e-02

0.000000 -0.000000 0.053580 **-5.358020e-02**

-0.144628 0.144628 0.093660 5.096788e-02

-0.289256 0.289256 0.208234 8.102205e-02

-0.433884 0.433884 0.380304 **5.358020e-02**

-0.549866 0.549866 0.541096 8.770416e-03

-0.665849 0.665849 0.700986 -3.513718e-02

-0.781831 0.781831 0.835411 **-5.358020e-02**

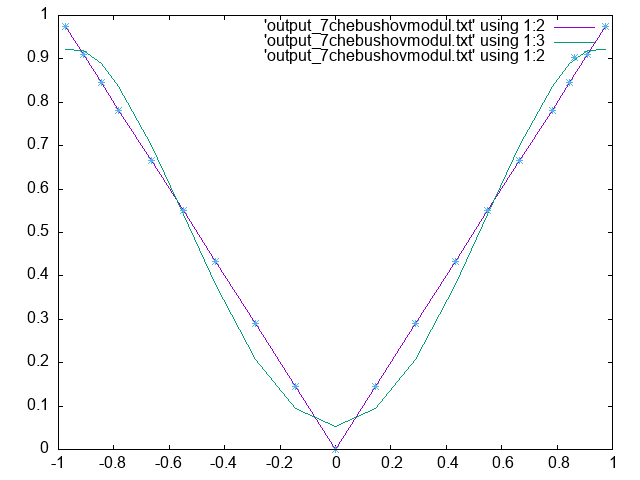
-0.846197 0.846197 0.888468 -4.227175e-02

-0.910562 0.910562 0.919136 -8.573788e-03

-0.974928 0.974928 0.921348 **5.358020e-02**

Goodbuy!

Посмотрим на графики исходной функции и полученного многочлена



Посмотрим график зависимости h от n

