Практические задачи по вычислительной математике.

Выполнил Рыжов Владислав, 714гр.

Код к каждой задачи прилагаю отдельно, некоторые задачи содержат несколько файлов.

Задача 1.1. Напишите подпрограмму, вычисляющую указанную функцию путем суммирования части ее ряда Маклорена:

1б) функцию ошибок $erf(x) = \frac{2}{\sqrt{\pi}} \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n+1}}{n!(2n+1)}$,

Далее, используя эту подпрограмму и формулу численного дифференцирования, найдите производную функции в точке x =0,5, 1, 5, 10, 30 с заданной точностью $\varepsilon = 10^{-6}$. Используйте несколько методов (сравните!), проведите вычисления с разными длинами мантиссы, сравните эффективность предложенных Вами методов. Исследуйте влияние ошибок округления.

Вот результат суммирования исходного ряда:

```
x=0.10000000000000000555
Exact:  0.11246291601828489748
Approximate(computated):    0.11246291601306945829
Error:  0.00000000000521543919

x=0.20000000000000001110
Exact:  0.22270258921047847434
Approximate(computated):    0.22270258655321062413
Error:  0.00000000265726785020

x=0.30000000000000004441
Exact:  0.32862675945912744879
Approximate(computated):    0.32862676095442994173
Error:  0.00000000149530249294

x=0.40000000000000002220
Exact:  0.42839235504666844934
Approximate(computated):    0.42839235425339672325
Error:  0.00000000079327172609

x=0.50000000000000000000
Exact:  0.52049987781304651868
Approximate(computated):    0.52049986354030419733
Error:  0.00000001427274232135

x=0.59999999999999997780
Exact:  0.60385609084792590817
Approximate(computated):    0.60385609759675140662
Error:  0.00000000674882549845

x=0.69999999999999995559
Exact:  0.67780119383741843642
Approximate(computated):    0.67780119018643469886
Error:  0.00000000365098373756

x=0.79999999999999993339
Exact:  0.74210096470766040433
Approximate(computated):    0.74210092986718356745
Error:  0.00000003484047683688

x=0.89999999999999991118
Exact:  0.79690821242283205184
Approximate(computated):    0.79690823301300972581
Error:  0.00000002059017767397

x=0.99999999999999988898
Exact:  0.84270079294971478312
Approximate(computated):    0.84270077928364151010
Error:  0.00000001366607327302

x=1.09999999999999986677
Exact:  0.88020506957408162219
Approximate(computated):    0.88020497006296516407
Error:  0.00000009951111645812

x=1.19999999999999995559
Exact:  0.91031397822963533439
Approximate(computated):    0.91031405150334321341
Error:  0.00000007327370787902

x=1.30000000000000004441
Exact:  0.93400794494065242368
Approximate(computated):    0.93400788564002845682
Error:  0.00000005930062396686
```

Результат суммирования продифференцированного ряда:

```
x=0.50000000000000000000
Exact:  0.87878257893544486912
Approximate(computated):        0.87878259218483145698
Error:  0.00000001324938658787

x=1.00000000000000000000
Exact:  0.41510749742059477319
Approximate(computated):        0.41510752350225782470
Error:  0.00000002608166305151

x=5.00000000000000000000
Exact:  0.00000000001567086653
Approximate(computated):        0.00000092315847623801
Error:  0.00000092314280537147


x=10.00000000000000000000
Exact:  0.00000000000000000000
Approximate(computated):        9190256480795944051697254 4.000
Error:  91902564807959440516972544.00000000000000000000

x=20.00000000000000000000000000000000000000000000000000
Exact:  0.00000000000000000000000000000000000000000000000000
Approximate(computated):        -3878104687598081762807256734795436915887689046410417529086538630818123221190375393677215014686696653135476507287664432463042016991279371728832502246053773312.000
Error:  3878104687598081762807256734795436915887689046410417529086538630818123221190375393677215014686696653135476507287664432463042016991279371728832502246053773312.00000000000000000000000000000
```

Дальше считать не было смысла, т. к. видно, что ошибка округления растет очень быстро и приводит к большим погрешностям. Ошибка округления растет быстрее чем линейная зависимость.

При использовании чисел типа double вместо float, то есть чисел с двойной длиной мантиссы, точность значительно повышается, но из-за быстрого роста ошибок округления результат при больших x имеет большие погрешности.

Один из методов заключается в вычислении каждого члена ряда прямым перемножением и суммированием его с предыдущими.
Второй метод заключается в использовании рекуррентной зависимости между членами суммы, в результате чего количество операций, а следовательно и сложность алгоритма падают.(O(2n) instead of O(n*n)).

Задача 1.2 Дан ряд $\sum_{n=0}^{\infty} a_n$. Найти сумму ряда S аналитически. Вычислить значения частичных сумм ряда $S_N = \sum_{n=0}^{N} a_n$ и найти величину погрешности при значениях значениях $N = 10, 10^2, 10^3, 10^4, 10^5$. Представить зависимости числа верных цифр результата от N. (Указание: точное значение полной и частичных сумм ряда вычислить аналитически).

1.2.4	$\dfrac{48}{5(n^2+6n+8)}$

$$a_n = \frac{48}{5(n^2+6n+8)} = \frac{48}{5}\,\frac{1}{(n+2)(n+4)} = \frac{48}{5}\left[\frac{1}{2}\,\frac{1}{n+2} - \frac{1}{n+4}\cdot\frac{1}{2}\right]$$

$$a_n = \frac{24}{5}\left[\frac{1}{n+2} - \frac{1}{n+4}\right]$$

$$\sum_{n=0}^{\infty} a_n = \frac{24}{5}\left[\frac{1}{2} - \cancel{\frac{1}{4}} + \frac{1}{3} - \cancel{\frac{1}{5}} + \cancel{\frac{1}{4}} - \cancel{\frac{1}{6}} + \cancel{\frac{1}{5}} - \frac{1}{7} + \cancel{\frac{1}{6}} - \frac{1}{8} + \ldots + \ldots\right]$$

$$S_n = \frac{24}{5}\left[\frac{1}{2} + \frac{1}{3} - \frac{1}{n+4}\right] = \frac{24}{5}\left[\frac{5}{6} - \frac{1}{n+4}\right]$$

$$S_n \xrightarrow[n\to\infty]{} \frac{24}{5}\cdot\frac{5}{6} = 4$$

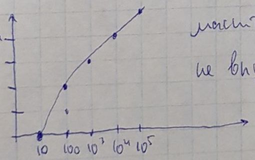
$$S_{10} = 4 - \frac{24}{5}\,\frac{1}{14} \approx 3{,}28 \qquad \Delta_{10} = \frac{24}{5}\cdot\frac{1}{14} \approx 0{,}34$$

$$S_{100} = 4 - \frac{24}{5}\,\frac{1}{104} \approx 3{,}91 \qquad \Delta_{100} = \frac{24}{5}\,\frac{1}{104} \approx 0{,}05$$

$$S_{10^3} = 4 - \frac{24}{5}\,\frac{1}{1004} \approx 3{,}9904 \qquad \Delta_{10^3} = \frac{24}{5}\,\frac{1}{1004} \approx 0{,}005$$

$$S_{10^4} = 4 - \frac{24}{5}\,\frac{1}{10004} \approx 3{,}99904 \qquad \Delta_{10^4} = \frac{24}{5}\,\frac{1}{10004} \approx 0{,}0005$$

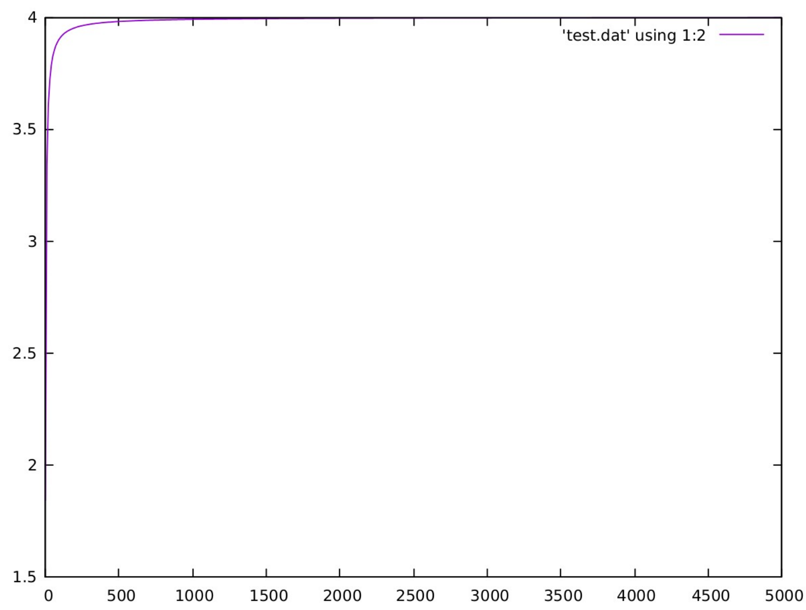
$$S_{10^5} = 4 - \frac{24}{5}\,\frac{1}{100004} \approx 3{,}999504 \qquad \Delta_{10^5} = \frac{24}{5}\,\frac{1}{100004} \approx 0{,}00005$$



масштаб не выдержан

Примечание

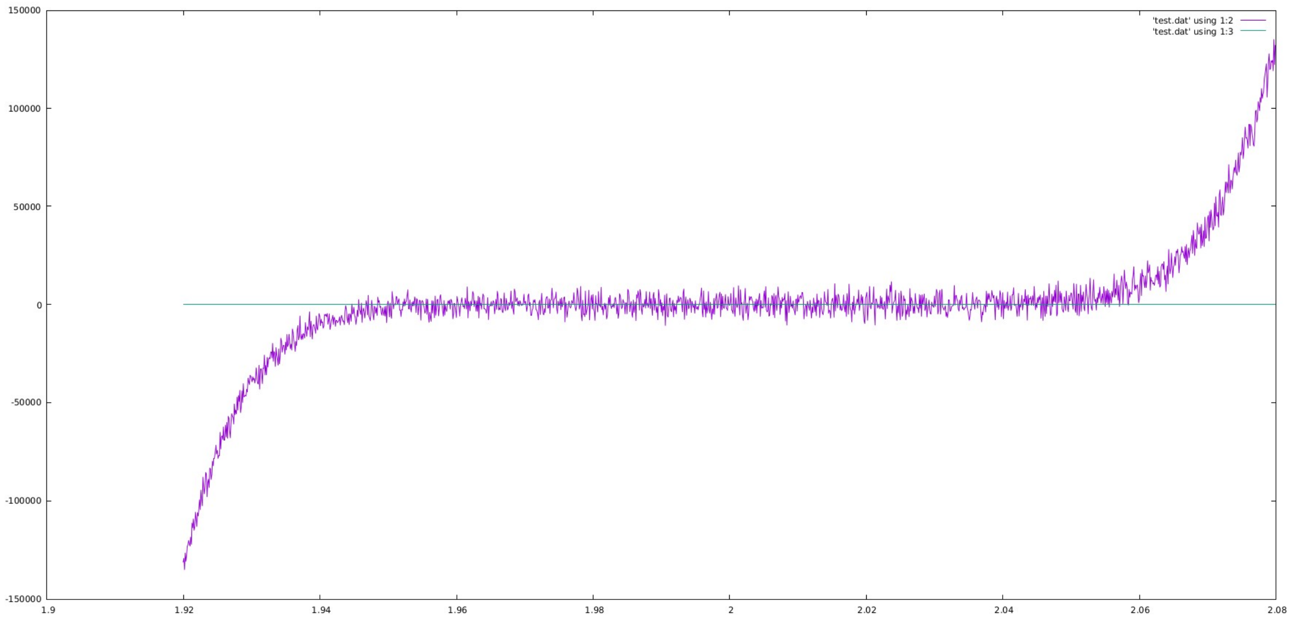
Округлял числа, чтобы не было зависимости $y=0$.



Задача 1.5 (I.9.2). Написать программу для вычисления многочлена $p(x) = \sum_{j=0}^{N} a_j x^j$, пользуясь схемой Горнера:

p = aN // for j = N – 1 to 0 // p = x·p + aj // end for // write x, p

для многочлена p(x) = (x – 2)9 на интервале [1.92, 2.08] с шагом 10–4. Результат нарисовать. Объяснить полученный результат. Сравнить его с вычислением по формуле p(x)=(x – 2)9. Почему алгоритм вычисления данного многочлена по схеме Горнера непригоден для численного определения нуля функции?



Как видно из графика, нуль многочлена довольно сложно определить, т. к. даже используя тип double(8 байт — максимум на 64-битной машине) мы сможем вычислить нуль в нашей задаче с абсолютной погрешностью = 0.05. Причина такой погрешности состоит в том, что при многократном перемножении чисел у нас появляется много цифр и последние теряются из-за ограниченности хранения в памяти.

Задача 1.6. (I.9.3.) Вычислить постоянную Эйлера

$$C = \lim_{n\to\infty}\left(\sum_{k=0}^{n}\frac{1}{k} - \ln n\right)$$

с точностью 10–10.

При выполнении данной задачи были применены оценки погрешности, взятые из научной статьи   https://research-journal.org/physics-mathematics/ob-analoge-postoyannoj-ejlera-maskeroni-i-zakonomernostyax-ego-izmeneniya/

$$H_n = \sum_{k=1}^{n} \frac{1}{k} = \ln n + \gamma + \alpha(n) \quad (2)$$

где H_n – гармоническое число, $\gamma = 0.557 \ldots$ – постоянная Эйлера-Маскерони, $a(n)$ – бесконечно малая функция. Отметим, что бесконечно малая функция, например, согласно [7, С. 4], имеет вид:

$$\alpha(n) = \frac{1}{2n} + \frac{1}{12n^2} + \frac{1}{120n^4} + \frac{1}{256n^6} \approx \frac{1}{2n} \quad (3)$$

Таким образом, a(n) можно ограничить сверху при помощи 1/n. То есть мы получили:

$$\sum_{k=1}^{n} \frac{1}{k} - \ln n - a(n) = \gamma$$

То есть наша погрешность не превосходит a(n) < 1/n. По условию a(n) < 10^(-10) достигается при n = 10^10.

Результат работы программы(которая работает довольно долго)

постоянная эйлера-маскерони равна:
0,5772156649