计算方法 B

Programming Assignment #1

2020.3.28

PB17000297 罗晏宸

Question 1

1 问题描述

分别对如下两个函数作编程计算

$$f(x) = \sqrt{x^2+9} - 3$$

$$g(x) = \frac{x^2}{\sqrt{x^2+9}+3}$$

分别取 $x = 8^{-1}, 8^{-2}, 8^{-3}, \cdots, 8^{-10}$，用单精度（即 `float` 型变量）进行计算，输出相应的函数值 $f(x)$ 和 $g(x)$，计算结果保留 12 位位数（用科学计数形式），比较并分析两种方法得到的计算结果。指出哪种方法得到的计算结果更可靠并给出理由或分析。

2 计算结果

使用 C++ 编程进行计算，得到结果如下：

x	$f(x) = \sqrt{x^2+9} - 3$	$g(x) = \frac{x^2}{\sqrt{x^2+9}+3}$
1.250000000000E-001	2.603054046631E-003	2.603037282825E-003
1.562500000000E-002	4.076957702637E-005	4.068982525496E-005
1.953125000000E-003	7.152557373047E-007	6.357827828651E-007
2.441406250000E-004	0.000000000000E+000	9.934107758625E-009
3.051757812500E-005	0.000000000000E+000	1.552204337285E-010
3.814697265625E-006	0.000000000000E+000	2.425319277008E-012
4.768371582031E-007	0.000000000000E+000	3.789561370325E-014
5.960464477539E-008	0.000000000000E+000	5.921189641133E-016
7.450580596924E-009	0.000000000000E+000	9.251858814270E-018
9.313225746155E-010	0.000000000000E+000	1.445602939730E-019

3 结果分析

直接观察上述计算结果，可以看到在 x 的值小于 8^{-3} 时，$f(x)$ 的计算结果已经失真，使用 Mathematica 进行高精度计算，在 200 位精度下得到可接受的精确值，并计算各数据的相对误差如下： 可以看到 $f(x)$ 的相对误差随着 x 值的减小而迅速增大，同时 $g(x)$ 的相对误差一直控制在 10^{-8} 这样一个比较好的数量级上，远小于前者。

x	$f(x)$ 的相对误差	$g(x)$ 的相对误差
1.250000000000E-001	-6.408111059344E-006	3.198320071447E-008
1.562500000000E-002	-0.001959919883E-000	7.294255084588E-008
1.953125000000E-003	-0.125000119209E-000	4.304788143774E-008
2.441406250000E-004	1.000000000000E+000	-3.145804955169E-008
3.051757812500E-005	1.000000000000E+000	-2.982813433699E-008
3.814697265625E-006	1.000000000000E+000	-2.980274579399E-008
4.768371582031E-007	1.000000000000E+000	-2.980234789005E-008
5.960464477539E-008	1.000000000000E+000	-2.980237333873E-008
7.450580596924E-009	1.000000000000E+000	-2.980233946459E-008
9.313225746155E-010	1.000000000000E+000	-2.980255563585E-008

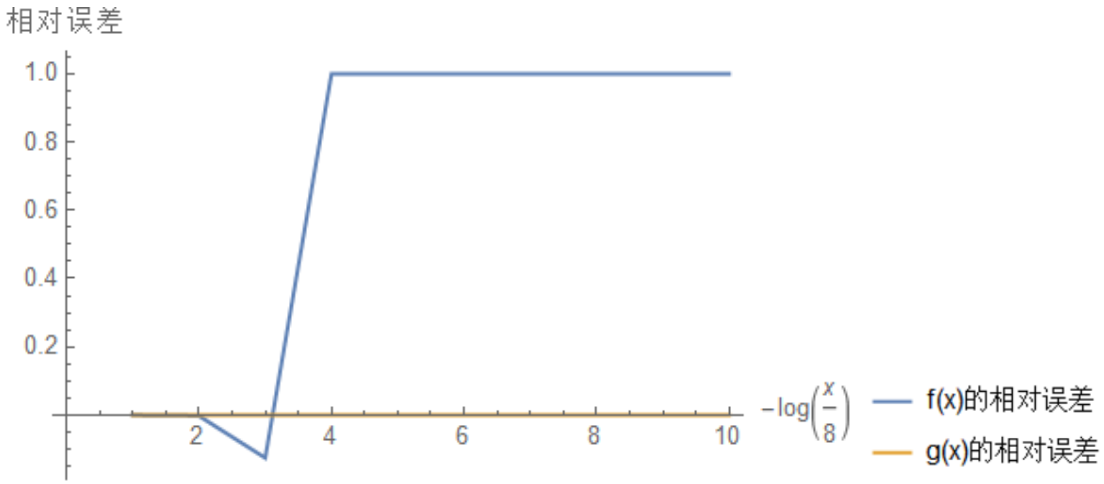

图 1: $f(x)$ 和 $g(x)$ 的相对误差关于 $-\lg\left(\frac{x}{8}\right)$ 的折线图

4 算法分析

由上可知，$g(x)$ 的计算方法能得到更可靠的结果，原因在于当 x 很小，以 $f(x)$ 的方式计算时，$\sqrt{x^2+9}$ 和 3 很接近，两者相减会增加相对误差；而以 $g(x)$ 的方式计算时，以 $\sqrt{x^2+9}+3$ 这样一个相对较大的数作为分母，可以避免绝对误差的增加。

5 实验结论

$f(x)$ 与 $g(x)$ 在数学上是相同两式，但是在实际的机器运算时，考虑到两个相近数相减时会导致相对误差变大，应当采取 $g(x)$ 进行计算，能够得到更可靠的结果。

Question 2

1 问题描述

给定一组数据

4040.045551380452, −2759471.276702747, −31.64291531266504,
2755462.874010974, 0.00005570529967428 93

分别采取以下 3 种方式求和：

(a) 顺序求和；

(b) 逆序（从后往前）求和；

(c) 正数从大到小求和，负数从小到大求和，再相加；

用双精度进行计算，计算结果至少保留 7 位有效数字（用科学计数形式）。比较 3 种方法得到的计算结果；指出哪种方法得到的计算结果更精确？试给出理由或分析。

2 计算结果

以三种方式计算得到的结果如下：

	方法 (a)	方法 (b)	方法 (c)
计算结果	1.025188136830E-010	-1.564330887049E-010	0.000000000000E+000

3 结果分析

可以看到方法 (c) 即“正数从大到小求和，负数从小到大求和，再相加”的计算结果已经失真了。方法 (a) 和方法 (b) 的计算结果在数量级上相同，但是有正负号的差别。使用 Mathematica 进行高精度计算，在 200 位精度下得到可接受的精确值是 $8.663428930000 \times 10^{-11}$，计算得到三种方法的相对误差为 可以看到方法 (b) 的相对误差已经达到 280%，甚至超过了失真的方法 (c)，

	方法 (a)	方法 (b)	方法 (c)
相对误差	-0.183351471009	2.805671749245	1.000000000000

相比之下方法 (a) 的相对误差较小，可以认为以此方法得到的结果较为准确。

4 算法分析

尽管所给数据的有效位数都未超过 `double` 类型的精度，但是中间结果受有效数字位数的限制，会在计算中出现截断误差。

方式 (a) 计算给定数据的序列如下

$$\begin{aligned} & 4040.045551380452 \\ - \quad & 2759471.276702747 \\ - \quad & 31.64291531266504 \\ + \quad & 2755462.874010974 \\ + \quad & 0.0000557052996742893 \end{aligned}$$

在前三步计算中受限于 `double` 类型 16 位有效数字的限制，分别出现了 3 或 5 位的有效数字丢失，产生了一定的截断误差。但注意到第三步运算的结果整数部分为 0，有效数字出现在小数部分，更接近结果的数量级 $10^{-}10$，因此截断误差较其余两种方式较小。

方式 (b) 计算给定数据的序列如下

$$\begin{aligned} & 0.0000557052996742893 \\ + \quad & 2755462.874010974 \\ - \quad & 31.64291531266504 \\ - \quad & 2759471.276702747 \\ + \quad & 4040.045551380452 \end{aligned}$$

由于第一个数是最接近运算结果的，其后的运算数据整数部分位数较多，因此结果的有效数字集中在小数点前，导致有至少 10 位有效数字的丢失。在随后的计算中除了最终结果，中间结果的整数部分均不小于 4 位数字，因此以这种方式运算得到的结果截断误差较大，较为不准确。

方式 (c) 先分别计算了正数与负数之和，由于结果是一个很小的数，这两者运算时实质上是两个相近的大数相减。这两者的整数部分均为 7 位，因此结果小数部分的有效部分最多只有 9 位，而精确值的数量级为 $10^{-}10$，即在精确值精度范围内的有效数字均在运算中丢失，因此出现了失真的结果 0。

5 实验结论

在实际的数值计算中，机器精度通常是一定的，在以预设的浮点数类型存储数据并进行数据之间的运算时，往往会出现截断误差，为了提高结果的精确性，应当避免在运算中出现和最终结果数量级相差较大的中间数据，以尽量维持在结果有效数字范围内的数据准确。