CS/SE 4X03 — Assignment 1

21 September, 2021

Due date: 1 October

Instructions

- If you write your solutions by hand, please ensure your handwriting is legible. We may subtract marks for hard-to-read solutions.
- Submit to Avenue a **PDF file** containing your solutions and the **required** MATLAB **files**.

  Assignments in other formats, e.g. IMG, PNG, **will not be marked**.

  Name your MATLAB files **exactly** as specified.
- Name your PDF file **Lastname-Firstname-studentnumber.pdf**.
- Submit **only what is required**.
- Do not submit zipped files. We will **ignore any compressed file** containing your files.

Problem 1 [2 points] Explain the output of this MATLAB code

```
a = -1.10714946411818e+17;
b = -8.039634349988262e-01;
c = 1.107149464118181e+17;
a+b+c
(a+b)+c
a+(b+c)
(a+c)+b
a+(c+b)
```

Problem 2 [7 points] Consider a floating-point (FP) system with 4 decimal digits and rounding to the nearest. Give an example for each of the following:

a. $(a+b)/2 \notin [a,b]$

b. $a + (b + c) \neq (a + b) + c$

c. $a * (b * c) \neq (a * b) * c$

where a, b, c are FP numbers in this system.

[4 points] Let a and b be IEEE-754 FP numbers and $a \leq b$. Can $(a+b)/2 \notin [a, b]$ occur? Explain.
Hint: for $x \leq y$, $\mathrm{fl}\,(x) \leq \mathrm{fl}\,(y)$.

Problem 3 [8 points] Consider the series expansion of e^x:

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots .$$

Write MATLAB functions as follows

`function s = expsum1(x)` computes an approximation to e^x by adding terms until the sum does not change.

`function s = expsum2(x)` returns `expsum1(x)` for $x \geq 0$ and 1/`expsum1(−x)` for $x < 0$.

`function s = expsum3(x)` accumulates the positive and negative terms separately and then adds the corresponding sums.

Write a main program `main_expsum.m` that calls these functions for

$$x = -20, -15, -5, -1, 1, 5, 15, 20$$

and outputs for each function results in the form

```
   x       accurate value      approx. value     abs. error  rel. error
 -20.0   2.061153622439e-09   6.138259738609e-09    4.08e-09   1.98e+00
 -15.0   3.059023205018e-07   3.059300523747e-07    2.77e-11   9.07e-05
   .
   .
   .
  20.0
```

(If y is an accurate value and $\widetilde{y}$ is an approximation, the absolute (abs.) error is $|y - \widetilde{y}|$ and the relative (rel.) error is $|y - \widetilde{y}|/|y|$.)

a. [6 points] Explain the accuracy of `expsum1`, `expsum2`, `expsum3`.

b. [2 points] Can `expsum3` produce accurate results for $x < 0$? Explain.

Submit

- Your MATLAB files to Avenue.
- PDF should contain MATLAB code, output, and discussion.

Problem 4 [6 points] We are not going to do C/C++ in this course, but this is a good exercise to learn about benchmarking and performance.

Read about

- the LINPACK benchmark, `https://en.wikipedia.org/wiki/LINPACK_benchmarks`.
- `long double` and `__float128`, `https://en.wikipedia.org/wiki/Long_double`

I have modified a bit the original benchmark, see the file `linpack.cc`. In Linux, save the given files `linpack.cc` and `makefile` and type `make`. This should produce the output file `benchmark.out`.

- Modify `linpack.cc`, see lines 48 and 52, so it can produce results for these two datatypes.
- Then uncomment in the `makefile`

```
TESTS = single double #longdouble float128
all: ${TESTS}
	./single > benchmark.out
	./double >> benchmark.out
	#./longdouble >> benchmark.out
	#./float128 >> benchmark.out
```

Executing `make` should produce results for the four data types.

Submit in the PDF

- [4 points] the output `benchmark.out`
- [2 points] discussion about the performance of these data types.

Problem 5 [10 points] Given a scalar function f, from

$$f(x+h) = f(x) + f'(x)h + O(h^2),$$

we can approximate the first derivative

$$f'(x) \approx g_1(x,h) := \frac{f(x+h) - f(x)}{h},$$

where the error in this approximation is $O(h)$.

From

$$f(x \pm h) = f(x) \pm f'(x)h + \frac{f''(x)}{2}h^2 + O(h^3),$$

we can approximate

$$f'(x) \approx g_2(x,h) := \frac{f(x+h) - f(x-h)}{2h}$$

where the error is $O(h^2)$.

a. [4 points] Let $f(x) = xe^x$ and $x_0 = 1$. Write a MATLAB program that computes the errors $|f'(x_0) - g_1(x_0, h)|$ and $|f'(x_0) - g_2(x_0, h)|$ for each $h = 10^{-k}$, $k = 1, 2, \dots, 16$.

Using `loglog`, plot on the same plot these errors versus h.

b. [2 points] For what h is the minimum of the errors achieved in each of these two approximations?

c. [4 points] Explain the behaviour of these errors. In particular, discuss the role of the cancellation and truncation errors.

Submit

- Avenue: your MATLAB code. Provide a main program with name `main_deriv.m` such that when it is executed it produces the required plot and the two values for h.
- PDF: plot, the two values for h, and explanations.

Problem 6 [6 points] Consider three methods for computing the sum of n floating-floating point numbers:

a. summation is in decreasing order of magnitude

b. summation is in increasing order of magnitude

c. using Kahan's summation algorithm
see `https://en.wikipedia.org/wiki/Kahan_summation_algorithm`

Apply these three methods to compute the sum $\sum_{i=1}^{n=10000} 1/i$. How would you compute an accurate result for this sum? Using such a result, report the errors in a., b., c.. in the form

decreasing order	error
increasing order	error
Kahan's sum	error

(Report in the format `format short e`)

Name your main program `main_sum.m`. When executed, it should produce the above output.

Submit

- Avenue: your MATLAB code.
- PDF: your MATLAB code, the output as above, and discussion.