MACM 316 – Computing Assignment #1

Due Date: Friday September 22 at 11:00pm.

Instructions: You must upload a single .pdf file consisting of 2 pages: page #1 is your report (containing all discussions, data and figures), and page #2 is a listing of your Matlab code. The assignment is due at **11:00 sharp!** If Crowdmark indicates that your submission was late, you will be assigned a grade of zero with no exceptions – so don't submit at the last minute. Your will receive an e-mail from Crowdmark containing a link that will allow you to upload your completed assignment, so remember to keep a copy of this message.

- Carefully review the **"Guidelines for Computing Assignments"** posted on Canvas.
- Acknowledge any collaborations or assistance from fellow students, TAs or instructor.
- If you have any questions about this assignment or Matlab, then you can obtain help in the computational workshops, tutorials, or the "Computing Assignment" discussion group on Canvas.

CA1 – Floating Point Errors and Matlab Plotting

This assignment is a more extensive investigation of the rounding error example we studied in class using the Matlab code `roundex.m` (posted on Canvas). The polynomial function $(x-a)^n$ with a any real number and n a positive integer can be written in expanded form as

$$(x-a)^n = \sum_{k=0}^{n} \binom{n}{k} x^{n-k}(-a)^k = x^n - \binom{n}{1} x^{n-1}a + \binom{n}{2} x^{n-2}a^2 - \binom{n}{3} x^{n-3}a^3 + \ldots + \binom{n}{n}(-a)^n \qquad (*)$$

where $\binom{n}{k} = \frac{n!}{(n-k)!\,k!}$ are binomial coefficients.

1. Plot $f(x) = (x-2)^n$ for $n = 1, 2, 3, 4, 5, 6$ on the domain $x \in [0, 4]$, using the factored (unexpanded) form. Display your 6 curves together on a single plot, and use different colors, line styles and/or a legend to clearly identify the various curves. Choose appropriate limits for the y-axis that ensure important features of the function are visible. Consider this your "exact result".

2. Write a Matlab function in an m-file named `fexpand.m` that computes the polynomial in the expanded form $(*)$. The first line of your code should be this function definition statement:

```
function fx = fexpand(a, n, x)
```

   The function has 3 input parameters (a, n and x) and returns a single output argument (the computed value of $f(x)$). You can make use of the built-in Matlab function `nchoosek` to compute the binomial coefficients. Test your code completely, making sure that it is able to exit gracefully with a suitable error/warning message for any values of the input arguments that might generate an invalid result.

3. Produce 6 plots depicting your "expanded" $f(x)$ curves that "zoom in" near the point $x = 2$ using a series of successively smaller x-intervals, $x \in [2-\delta,\ 2+\delta]$ for $\delta = 0.5, 0.1, 0.05, 0.025, 0.01, 0.005^{\dagger}$. In your report, you should only include the plots for $\delta = 0.5, 0.05, 0.005$, with two select values of the polynomial degree n, for a total of $3 \times 2 = 6$ plots – and choose the two n values that you think best illustrate your conclusions! Describe any interesting behaviour or differences that arise between the various n, δ values. In addition, be sure to comment on the following:

   (a) Identify the smallest value of the exponent n for which your plots of the expanded polynomial $(*)$ differ significantly from the "exact" plots (in part 1).

   (b) Describe what happens as n increases (it might help to try a few n even larger than 6).

   (c) Explain why performing these computations in double-precision floating point arithmetic gives rise to the errors you observe. Be specific!

---

† Note that you should choose your plotting points to be consistent with the size of each x-interval! In other words, as δ gets smaller, you should also decrease the spacing between plotting points so that you can clearly resolve any interesting features.