Problem 1. (a) According to a standard trigonometric identity, the following two expressions are equal in exact arithmetic:

\[ x_1 = \cos(2 + \delta) - \cos(2 - \delta) \]
\[ x_2 = -2 \sin(2) \sin(\delta). \]

Use Matlab or other numerical software to evaluate approximations \( \hat{x}_1 \) and \( \hat{x}_2 \) in double-precision arithmetic for \( \delta = 10^{-n} \), \( n = 4, 5, 6, 7, 8 \). For each choice of the integer \( n \) how many significant figures do \( \hat{x}_1 \) and \( \hat{x}_2 \) have in common?

(b) Which of the two approximations \( \hat{x}_1 \) or \( \hat{x}_2 \) do you believe is closer to the exact result \( x_1 = x_2 \), and why? Is either one accurate to double precision?

(c) How many significant figures are there in IEEE standard floating point double precision arithmetic? Using this result, explain the magnitude of the relative errors observed in part (a).

(d) Now consider a third approximation

\[ x_3 = -2 \sin(2) \delta \]

which satisfies \( \lim_{\delta \to 0} (x_3/x_2) = 1 \) in exact arithmetic. Again use Matlab or other numerical software to evaluate approximations \( \hat{x}_3 \) and \( \hat{x}_3 \) in double-precision arithmetic for \( \delta = 10^{-n} \), \( n = 4, 5, 6, 7, 8 \). For each choice of the integer \( n \) how many significant figures do \( \hat{x}_2 \) and \( \hat{x}_3 \) have in common?

(e) Explain the magnitude of the relative errors observed in part (d) for all values of the integer \( n \) and in particular for the case \( n = 8 \).
Problem 2. a) Use Taylor series with remainder to prove that

\[ \frac{f(x + h) - f(x - h)}{2h} = f'(x) + \frac{1}{6}f'''(\xi)h^2 \]

for \( f \in C^3 \) and some \( \xi \) between \( x - h, x + h \). You will need to use also the Intermediate Value Theorem to get the form of the remainder stated above.

(b) Use part (a) to estimate the total error in approximating the derivative \( f'(x) \) with the difference quotient

\[ \frac{\hat{f}(x + h) - \hat{f}(x - h)}{2h} \]

where \( \hat{f}(x) \) is the function evaluated in a computer arithmetic to relative precision \( \epsilon \). Derive an upper bound involving \( h, \epsilon, M_0 = \max_x |f(x)| \) and \( M_3 = \max_x |f'''(x)| \).

(c) Find the value \( h = h_* \) to give the smallest error bound in (b). Using this optimal value in double precision arithmetic, what is the approximate number of digits of accuracy that can be expected in the estimate of \( f'(x) \)? Assume that the quantities \( M_0, M_3 \) are of order unity.

(d) For the function \( f(x) = e^x \) evaluate the approximation in part (b) for \( x = 0 \) using double-precision arithmetic and \( h = 10^{-n} \), for integers \( 1 \leq n \leq 10 \) and calculate the error in this approximation of \( f'(0) \). Plot error versus \( n \) using \texttt{semilogy} in Matlab. Does the plot agree with the error bounds and optimal value \( h_* \) in part (c)? Explain.
Problem 3. (a) Suppose that \( f \in C[0,1] \), i.e. that \( f \) is a continuous function of \( x \) over the closed interval \( 0 \leq x \leq 1 \). Consider the problem of finding the anti-derivative function \( F \in C^1[0,1] \), i.e. the element \( F(x) \) in the space of continuously differentiable functions such that \( F'(x) = f(x) \). Does \( F \) exist? Is \( F \) unique? If not, what additional condition is required to make the anti-derivative \( F \) unique?

(b) Show for any function \( f_N \in C[0,1] \) approximating \( f \) that
\[
\max_{x \in [0,1]} |F_N(x) - F(x)| \leq \max_{x \in [0,1]} |f_N(x) - f(x)|. 
\]
Is the problem of finding the anti-derivative well-posed? Explain your answer.

(c) Suppose now that \( f \in C^1[0,1] \). Show that the function defined by
\[
f_N(x) = f(x) + \frac{1}{N} \sin(2\pi N^2 x)
\]
is also in \( C^1[0,1] \) for all integers \( N \) and
\[
\max_{x \in [0,1]} |f(x) - f_N(x)| = \frac{1}{N}
\]
but
\[
\max_{x \in [0,1]} |f'(x) - f'_N(x)| = 2\pi N.
\]

(d) Is finding the derivative \( f' \in C[0,1] \) of a function \( f \in C^1[0,1] \) a well-posed problem? Explain in what sense it is or is not.
Problem 4. In this problem we consider two relatively complicated functions, the sine integral function

\[
(i) \quad f(x) = \text{Si}(x) := \int_0^x \frac{\sin t}{t} \, dt
\]

which is given in Matlab by \texttt{sinint} and the function

\[
(ii) \quad f(x) = \sin(x) \sin(2x) \sin(3x) \sin(5x) \sin(7x).
\]

We shall invert these functions by solving the equation \( f(x) = y \) for particular values of \( y \) using both the Newton method and secant method, and compare their efficiency.

(a) Rewrite the Matlab script \texttt{newton.m} from the course website with error tolerance \( tol = 10^{-15} \) and maximum iterations \( maxit = 1000 \) as a function file in the format

\[
[x, ee] = \text{newton}(\Phi(x), F(x), DF(x), x0)
\]

with function \( F \), its derivative \( F' \), and an initial guess \( x_0 \) as inputs, and the vector of successive iterates \( x = (x_0, x_1, \ldots, x_N) \) and the corresponding vector of error estimates \( e = (|x_0 - x_{-1}|, |x_1 - x_0|, \ldots, |x_N - x_{N-1}|) \) as outputs (for \( x_{-1} = 0 \)). Likewise, rewrite the script \texttt{secant.m} as a function file in the format

\[
[x, ee] = \text{secant}(\Phi(x), F(x), a, b)
\]

with \( x_0 = a \) and \( x_1 = b = a - F(a)/F'(a) \) as the two initial guesses.

(b) For function (i), use Newton's method and secant method with \( F'(y) = f(y) - y \) for \( y = 1 \) to find the value \( x_* \) such that \( f(x_*) = \text{Si}(x_*) = 1 \). Set \( x_0 = 1 \) for both methods. For secant method, generate the second guess \( x_1 = b \) by one Newton iteration. For each method, record the iterates, the estimated errors, and the wall clock time. To estimate the latter accurately, average over 100 trials using the loop

\[
\text{num}=100
\]

\[
\text{ii}=1: \text{num}
\]

\[
\text{tic}
\]

\[
[x, ee] = \text{newton}(\Phi(x), F(x), DF(x), x0);
\]

\[
\text{timen} = \text{timen} + \text{toc};
\]

\[
\text{tic}
\]

\[
b = a - F(a)/F'(a);
\]

\[
[x, ee] = \text{secant}(\Phi(x), F(x), a, b);
\]

\[
\text{times} = \text{times} + \text{toc};
\]

\[
\text{timen} = \text{timen} / \text{num};
\]

\[
\text{times} = \text{times} / \text{num};
\]

Which method requires fewer iterations? Which method requires a smaller wall clock time? Explain your results in terms of the clock time required to evaluate each of the functions \( F(x) \) and \( F'(x) \) for one value of \( x \).

(c) Repeat part (b) for function (ii), \( y = 0.1 \), and \( x_0 = a = 0.25 \).