Problem 1 [4 points] Suppose you enter three numbers x, y, and z from the keyboard of your computer, store them in double precision variables, and compute `x*y/z`. Assuming that this expression is evaluated in double precision, calculate a bound for the relative error in the computed result.

Problem 2 [4 points] Consider $f(x) = (e^{2x} - 1)/(2x)$. Let $x = 1\text{e}{-}10$ and assume double precision.

(a) When evaluated in double precision, **exp**`(2*x)` is `1.000000000200000`. Without using the **exp** function, how would you obtain this value?

(b) Describe an approach for computing $f(x) = (e^{2x} - 1)/(2x)$ such that loss of significance is avoided when x is near zero.

(c) Using your approach, what would you obtain with $x = 1\text{e}{-}10$?

Problem 3 [4 points] Show how to rewrite the following expressions to avoid cancellations for the indicated arguments:

a. (1 point) $\sqrt{x+1} - 1, \quad x \approx 0$

b. (1 point) $\sin x - \sin y, \quad x \approx y$

c. (1 point) $x^2 - y^2, \quad x \approx y$

d. (1 point) $(1 - \cos x)/\sin x, \quad x \approx 0$

Problem 4 [4 points] Suppose $\cos x$ is approximated by an interpolating polynomial of degree n using $(n+1)$ equally spaced points in the interval $[0, 2\pi]$.

(a) How accurate is this approximation in terms of n?

(b) What is the minimum number of points needed to achieve error less than 10^{-6}?

Problem 5 [2 points]

(a) Approximate $\int_1^3 1/x^2 dx$ using the midpoint rule with equally spaced points with $h = 0.5$.

(b) Compute a bound for the error in this approximation.

Problem 6 [4 points] Let x and y be finite IEEE-754 double precision numbers. Consider the evaluation of

```
u = x*sqrt(y);
v = 2*u;
```

in IEEE double precision (`sqrt` is the square root function). Assume no overflows or underflows occur in this evaluation. Derive a bound for the relative error in v. This bound should be a numeric value.

Problem 7 [4 points] Given the three data points $(-1,1)$, $(0,0)$, $(1,1)$ write the polynomial interpolating them using

(a) monomial basis

(b) Lagrangian basis

(c) Newton basis

Show that the three representations give the same polynomial.

Problem 8 [3 points] Consider

$$f(x) = (x-1)^2 e^x = 0.$$

(a) Write Newton's method when applied to this f.

(b) Show that when the initial condition is $x_0 \approx 1$ Newton converges linearly.

(c) What would the bisection method produce on the interval $[-1,2]$?

Problem 9 [4 points] If A and B are $n \times n$ matrices, with A nonsingular, and c is a vector with n elements, how would you efficiently compute the product $A^{-1}Bc$? What is the complexity of your computation and why?

Problem 10 [3 points] Consider the following method for integrating $y' = f(t,y)$, $y(t_0) = y_0$:

$$y_{i+1} = y_i + \frac{h}{2}[f(t_i, y_i) + f(t_{i+1}, y_{i+1})].$$

(a) (1 point) Consider $y' = -t\sin(y)$. Write the above formula when applied to this ODE.

(b) (2 points) To solve for y_{i+1} we need to apply Newton's method. Write this method for solving for y_{i+1}. That is, write the function f and its derivative f' in

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

Problem 11 [5 points]
If A, B, and C are $n \times n$ matrices, with B and C nonsingular, and b is an n-vector, how would you implement the formula

$$x = B^{-1}(2A+I)(C^{-1}+A)b$$

without computing any matrix inverses? (Do not write Matlab code, just give a sequence of formulas.) Here I is the $n \times n$ identity matrix.

Problem 12 [5 points]

(a) (1 points) Approximate $\int_1^2 x\ln(x+2)dx$ using Simpson's rule.

(b) (2 points) How many (equal) subintervals n are needed to approximate this integral with the composite trapezoid rule such that the error in the approximation is $\leq 10^{-10}$.

(c) (2 points) After you have obtained your n, describe how you would verify that this n gives you an error within $\leq 10^{-10}$.

Problem 13 [4 points] A planet follows an elliptical orbit, which can be represented in a Cartesian (x,y) coordinate system by the equation

$$ay^2 + bxy + cx + dy + e = x^2$$

Suppose you are given coordinates (x_i, y_i), $i = 1, \ldots, n$.
Describe how you would compute the constants a, b, c, d, e in this equation. That is, setup a problem and explain how you can solve it.

Problem 14 [9 points] Given n double precision numbers $a_1,\ a_2,\ \ldots,\ a_n$, consider evaluating the expression $\sqrt{a_1^2+a_2^2+\cdots+a_n^2}$ in double precision. An overflow or underflow may occur. Show how to evaluate this expression such that an overflow is avoided. The largest representable number in double precision is $\approx 1.7976931348623157 \times 10^{308}$.

Problem 15 [4 points] The following Matlab program

```
y = [ pi, 1e-8, 1e-100, 1e-155];
for i = 1:length(y)
  x = y(i);
  u = 1.0/x^3;
  v = sin(x*x)^3;
  A = 1.0/sqrt(v) - u;
  fprintf("i=% d, x = %g\n",i, x);
  fprintf("    A = %g\n", A);
end
```

outputs

```
i= 1, x = 3.14159
   A = -0.0322515
i= 2, x = 1e-08
   A = 0
i= 3, x = 1e-100
   A = Inf
i= 4, x = 1e-155
   A = NaN
```

In the following, provide sufficient detail justifying your answer.

(a) When $i=1$, is the value of `A` accurate?

(b) When $i=2$, why `A = 0`?

(c) When $i=3$, why `A =` **Inf**?

(d) When $i=4$, why `A =` **NaN**?