

Problem 1 [8 points] Matlab's $\text{fma}(a,b,c)$ function computes $a*b+c$, where the mathematically correct result of $a*b+c$ is rounded to the nearest. That is, $\text{fl}(a*b+c) = (a*b+c)(1+\delta)$, where $|\delta| \leq u$, and u is the unit roundoff.

The following Matlab code

```
for i=5:15
    x = 10^i;
    y = fma(x,x,-x*x);
    fprintf("x=%.1e y=% e\n", x, y);
end
```

outputs

```
x=1.0e+05 y= 0.000000e+00
x=1.0e+06 y= 0.000000e+00
x=1.0e+07 y= 0.000000e+00
x=1.0e+08 y= 0.000000e+00
x=1.0e+09 y= 0.000000e+00
x=1.0e+10 y= 0.000000e+00
x=1.0e+11 y= 0.000000e+00
x=1.0e+12 y= 1.677722e+07
x=1.0e+13 y=-4.764729e+09
x=1.0e+14 y= 4.168803e+11
x=1.0e+15 y=-1.988462e+13
```

- [2 points] Explain why for $x = 10^k$, $k = 5, 6, \dots, 11$, y is 0 in the output.
- [2 points] Explain why the rest of the y results are not 0.
- [4 points] For a given $k \geq 12$ and $x = 10^k$, derive an upper bound for the absolute value of y , $\text{abs}(y)$.

Problem 2 [2 points] Let x and y be exactly representable machine numbers. That is, $\text{fl}(x) = x$ and $\text{fl}(y) = y$. When $x \approx y$, to evaluate $\sin(x) - \sin(y)$, is

$$2 \cos \frac{x+y}{2} \sin \frac{x-y}{2}$$

a more suitable formula? Justify your answer.

Problem 3 [3 points] Consider the function

$$f(x) = \frac{\sin(\pi/4 + x) - \cos(\pi/4 + x)}{x}.$$

$$\text{As } x \rightarrow 0, f(x) \rightarrow \sqrt{2}. \quad (1)$$

(You do not have to prove (1)). Suppose you try to check numerically if (1) is true and execute the following code

```
f = @(x) (sin(pi/4+x)-cos(pi/4+x))./x;
for i=11:25
    x = 10^(-i);
    fprintf("%.0e    % e    % e\n", x, f(x), abs(f(x)-sqrt(2)))
end
```

It produces

1e-11	1.414213e+00	3.713753e-07
1e-12	1.414091e+00	1.224959e-04
1e-13	1.414424e+00	2.105710e-04
1e-14	1.409983e+00	4.230321e-03
1e-15	1.443290e+00	2.907637e-02
1e-16	1.110223e+00	3.039905e-01
1e-17	-1.110223e+01	1.251644e+01
1e-18	-1.110223e+02	1.124365e+02

- [1 point] As x gets smaller, the error $|f(x) - \sqrt{2}|$ increases. What is the reason for this increase?
- [2 points] Suggest a more accurate formula for evaluating $f(x)$ when $x \approx 0$. You can use

$$\sin(a+b) = \sin(a)\cos(b) + \sin(b)\cos(a)$$

$$\cos(a+b) = \cos(a)\cos(b) - \sin(a)\sin(b)$$

$$\sin(\pi/4) = \cos(\pi/4) = 1/\sqrt{2}.$$

- Bonus [4 points] Explain why exactly these numbers are produced: 1.110223e+00, -1.110223e+01, -1.110223e+02.

Problem 4 [5 points]

- a. [2 points] Given the values

x	-1	-0.5	0.5	1
$f(x)$	-1	1	-1	1

find the polynomial $p(x)$ of degree three that interpolates $f(x)$.

- b. [3 points] Assume
- $\max_{x \in [-1,1]} |f^{(4)}(x)| \leq 1$
- . Find the value of the smallest
- M
- such that

$$\max_{x \in [-1,1]} |f(x) - p(x)| \leq M.$$

Problem 5 [6 points] Consider the function $f(x) = 1 + x \sin(x) - e^{-x/2}$. You are given the values

x	$1 + x \sin(x)$	$e^{-x/2}$
-0.9	1.704994	1.568312
-0.7	1.450952	1.419068
-0.5	1.239713	1.284025
-0.3	1.088656	1.161834
-0.1	1.009983	1.051271
0.1	1.009983	0.951229
0.3	1.088656	0.860708
0.5	1.239713	0.778801
0.7	1.450952	0.704688
0.9	1.704994	0.637628
1.1	1.980328	0.576950

- a. [1 point] Give an interval of length 0.2 containing a root of $f(x)$. Justify your answer.
- b. [1 point] Write Newton's method for finding a root of $f(x)$.
- c. [1 point] What initial guess would you use to start Newton's method. Justify your answer.
- d. [2 points] With your initial guess, how many iterations do you expect for Newton to converge in double precision so the error $|r - x_n| \leq 10^{-15}$. Here r denotes a root of $f(x)$ and x_n is the computed approximation at iteration n .
- e. [1 point] How many roots are in $[-0.9, 1.1]$? Justify your answer.

Problem 6 [7 points] A GPS receiver calculates its position by utilizing signals from four satellites. Each satellite i , $i = 1, 2, 3, 4$, transmits its coordinates (x_i, y_i, z_i) and the time t_i (according to its internal clock) when they are sent.

The GPS receiver records (t_i, x_i, y_i, z_i) and the time received, denoted by \hat{t}_i , as measured by its own clock. The time taken for the signal to travel from a satellite to the receiver is $\hat{t}_i - t_i + b$, where b is the difference in accuracy between the satellite clock and the GPS clock. A satellite clock is more accurate than the receiver clock, and b accounts for this disparity. Assuming synchronization among the satellite clocks, the value of b remains consistent across all satellites. The signals travel with the speed of light; denote this speed by the constant c .

- [4 points] Assuming $\hat{t}_i, t_i, x_i, y_i, z_i$, for $i = 1, 2, 3, 4$, are known, formulate the system of equations from which you can find the coordinates x, y, z of the GPS receiver's position. Note: b is unknown.
- [2 points] Derive the Jacobian of this system.
- [1 points] What Matlab function would you use to solve this system?

Problem 7 [5 points] Suppose you have values for m points (t_i, y_i) for $i = 1, 2, \dots, m$. You need to find the constants a, b , and c such that

$$y = ae^{bt+ct^2}$$

fits these data in a least squares sense.

- [4 points] Describe how you would find these constants.
- [1 points] How can you check that the fit you obtain is a “good one”?

Problem 8 [6 points] Consider the IVP

$$x'(t) = -(t+1)x^2 + 2, \quad x(0) = 1.$$

- [2 points] Calculate x at $t_1 = 0.1$ using the forward Euler's method.
- [2 points] Write the backward Euler's method to compute a value for x at $t_1 = 0.1$.
- [2 points] In backward Euler, what is the nonlinear equation to be solved? (You don't need to solve it.)
What initial value would you use in Newton's method?

Problem 9 [6 points] Consider implementing an adaptive quadrature based on the trapezoidal rule.

- a. [2 points] Show how you would estimate the error.
- b. [4 points] Write in pseudo code a recursive algorithm implementing the adaptive function $Q = \text{quad}(f, a, b, \text{tol}, n)$, which computes Q such that

$$\left| Q - \int_a^b f(x) dx \right| \leq \text{tol}.$$

- f is the function to be integrated over $[a, b]$.
- tol is tolerance.
- n is the maximum of depth the recursion.

Problem 10 [6 points] Let n be a large positive integer (e.g. $n = 10000$). Assume that

- A, B, C , are known $n \times n$ matrices, and A is nonsingular.
- b_1, b_2, b_3 are known n -vectors.

Consider solving

$$\begin{bmatrix} A & 0 & 0 \\ B & A & 0 \\ C & B & A \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

for the n -vectors x_1, x_2 , and x_3 .

- a. [3 points] Describe how you would solve this system efficiently.
- b. [3 points] Derive the complexity of your approach in big-O notation. It is not necessary to perform an accurate operation count, big-O notation is sufficient.

END OF EXAMINATION