ASEN 3502 Computational Methods
Practice Exam 1 (2026)

Instructions: This practice exam is meant to show the style and scope of Exam 1; it is not an exhaustive list of what may be asked. The equation sheet distributed with the exam will be available. On the real exam, you will write your answers on a provided answer sheet. For this exam, use your own paper.

Question 1. (20 points) A friend knows the Gaussian elimination algorithm well but has never heard of LU factorization.

1. In a few sentences, explain to them what LU factorization computes and how it relates to the steps of Gaussian elimination.
2. Describe a situation where LU factorization is preferable to Gaussian elimination, and explain why using big-O notation.

Question 2. (20 points) Count the floating point operations required by the following algorithm, which computes the matrix-vector product $\{y\} = [A]\{x\}$ for an $n \times n$ matrix $[A]$. Report your answer as an exact expression in n and use big-O notation to characterize the growth rate.

```
function matvec([A], {x})
    for i = 1 to n
        y_i = 0
        for j = 1 to n
            y_i = y_i + a_ij x_j
        end
    end
    return {y}
end
```

Question 3. (15 points) You have successfully executed a bisection algorithm starting from an initial interval that contains a root on continuous function $f(x)$. The algorithm outputs a final interval, $[x_l, x_u]$. Let the midpoint of the interval be $x_m = \frac{x_l + x_u}{2}$. Indicate whether each statement must be TRUE or may be FALSE. Justify your answers (Hint: for false answers, use a counterexample).

1. There must be exactly one root in the interval.

   ☐ TRUE ☐ FALSE

   Justification:

2. The following relation must be satisfied: $f(x_l) \leq f(x_m) \leq f(x_u)$.

   ☐ TRUE ☐ FALSE

   Justification:

3. Let $r = \frac{x_u - x_l}{2}$. There must be a root within the interval $[x_m - r, x_m + r]$.

   ☐ TRUE ☐ FALSE

   Justification:

Question 4. (15 points) Below is a partial Matlab implementation of the bisection method for finding a root of `f` on the interval `[x_l, x_u]`. Fill in the missing code in the box that updates the bracket so that it always contains a root.

```
function x_m = bisect(f, x_l, x_u, atol)
    while (x_u - x_l) > atol
        x_m = (x_l + x_u)/2;
        % ----- fill in the endpoint update below -----









        % ---------------------------------------------
    end
    x_m = (x_l + x_u)/2;
end
```

Question 5. (30 points) Consider $f(x) = x^3 - 2$.

1. Starting from $x_0 = 2$, carry out two iterations of Newton-Raphson by hand to find x_1 and x_2. Show your work and report at least three significant figures.
2. Compute the approximate percent relative error ε_a after the second iteration.
3. Draw a sketch showing how the algorithm executes on this function, similar to those drawn in class.
4. Give a starting point x_0 from which Newton-Raphson will fail to find a root, and explain what goes wrong.

Solutions

Question 1

(a) LU factorization writes $[A] = [L][U]$, where $[U]$ is the upper triangular matrix that Gaussian elimination produces during forward elimination, and $[L]$ is unit lower triangular with the elimination factors stored below the diagonal:

$$f_{ik} = \frac{a_{ik}}{a_{kk}}.$$

It is Gaussian elimination that remembers the multipliers instead of throwing them away, and it does not touch $\{b\}$.

(b) Solving many systems with the same $[A]$ but different $\{b\}$, e.g., many load cases on one structure. The $O(n^3)$ factorization is done once, and each new right-hand side costs only two triangular solves, $O(n^2)$. Gaussian elimination would repeat the $O(n^3)$ work for every $\{b\}$.

Question 2

Line	Statement	Flops	Times executed
3	$y_i = 0$	0	n
5	$y_i = y_i + a_{ij}\, x_j$	2	$n \cdot n = n^2$

$$\text{Total flops} = 0 \cdot n + 2 \cdot n^2 = 2n^2 = O(n^2)$$

Question 3

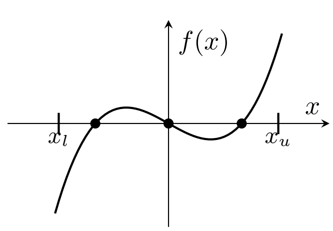
(a) False. Bisection only guarantees that $f(x_l)$ and $f(x_u)$ have opposite signs, which is also true when there are three (or any odd number of) roots. For example, $f(x) = x^3 - x$ on $[-2, 2]$ has three roots, and if the tolerance is loose bisection could stop with all three still inside the interval.

(b) False. Bisection guarantees a sign change, not that f is increasing. For example, $f(x) = -x$ on $[-1, 1]$ gives

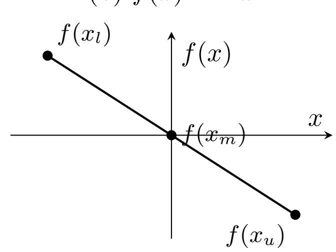
$$f(x_l) = 1 \;>\; f(x_m) = 0 \;>\; f(x_u) = -1.$$

(c) True. $[x_m - r, x_m + r] = [x_l, x_u]$ is exactly the final bracket. As long as the initial interval has $f(x_l)$ and $f(x_u)$ with different signs, bisection always keeps the sign change when it replaces an endpoint, so a root is always within the refined bracket.

(a) $f(x) = x^3 - x$



(b) $f(x) = -x$



Question 4

```
if sign(f(x_m)) == sign(f(x_u))
    x_u = x_m;
else
    x_l = x_m;
end
```

Alternative implementations such as checking `f(x_m)*f(x_l) < 0` are acceptable. The key is that the sign change is preserved: replace the endpoint whose function value has the same sign as $f(x_m)$.

Question 5

(a) With $f'(x) = 3x^2$,

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} = 2 - \frac{6}{12} = 1.5$$

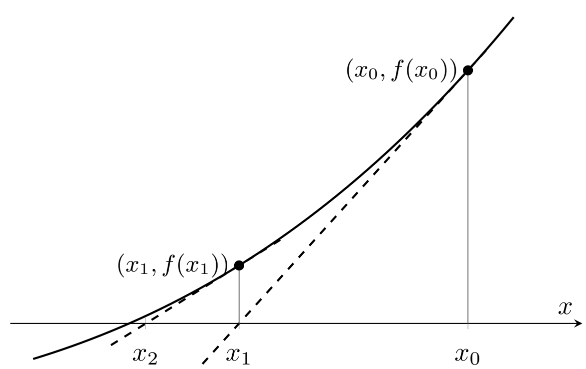
$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)} = 1.5 - \frac{1.375}{6.75} = 1.5 - 0.204 = 1.30$$

(The true root is $\sqrt[3]{2} = 1.260$.)

(b)

$$\varepsilon_a = \left|\frac{x_2 - x_1}{x_2}\right| \times 100\% = \left|\frac{1.30 - 1.5}{1.30}\right| \times 100\% = 15.7\%$$

(c) The tangent at $(2, 6)$ has slope 12 and crosses the axis at $x_1 = 1.5$; the tangent at $(1.5, 1.375)$ has slope 6.75 and crosses at $x_2 = 1.30$.



(d) $x_0 = 0$. There $f'(0) = 0$, so the tangent is horizontal and never crosses the axis. The update divides by zero (`Inf` or `NaN` in Matlab). Any x_0 close enough to 0 also causes trouble: the first step is thrown far from the root because the denominator is tiny.