MMAE 350 – Midterm 1 Study Guide

Scope of the Exam

This study guide is intended to help you organize your preparation for Midterm 1 and to highlight the types of concepts and skills you should be comfortable with. It is not a practice exam, and the questions listed here are meant to guide your studying rather than cover every possible topic. Midterm 1 covers material from Chapters 1 and 2 of the course text. The emphasis is on understanding fundamental numerical ideas, performing small hand calculations, and translating algorithms into clear pseudocode.

1 Python and Computational Foundations

Open-Ended Questions

1. What is a virtual environment, and why is it good practice to use one?
2. What role do Jupyter notebooks play in computational mechanics?
3. Explain the difference between a Python list and a NumPy array.
4. Why are loops and conditional statements essential for numerical algorithms?

Hand Calculations

1. Given the vector

   $$x = [1, 3, 5, 7],$$

   write out explicitly what is returned by:

   $$x[1:3], \quad x[::2].$$

2. Given the matrix

   $$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix},$$

   and vector

   $$b = \begin{bmatrix} 5 \\ 6 \end{bmatrix},$$

   write out the matrix–vector product Ax symbolically.

Pseudocode

1. Write pseudocode for a `for` loop that computes the squares of the integers from 1 to n.
2. Write pseudocode for an `if--else` block that checks whether a number is positive, negative, or zero.

2 Matrix Algebra

Open-Ended Questions

1. What does it mean for a matrix to be invertible?
2. Why is explicitly computing a matrix inverse usually discouraged in numerical work?
3. Explain the geometric meaning of the transpose of a matrix.
4. What is the physical meaning of a matrix–vector product in mechanics?

Hand Calculations

1. Compute the transpose of
$$A = \begin{bmatrix} 2 & -1 & 0 \\ 1 & 3 & 4 \end{bmatrix}.$$
2. Compute the determinant and inverse (by hand) of
$$A = \begin{bmatrix} 4 & -1 \\ 2 & 3 \end{bmatrix}.$$
3. Verify that $AA^{-1} = I$ for your result above.

Pseudocode

1. Write pseudocode for multiplying a matrix A by a vector x using nested loops.
2. Write pseudocode that counts how many entries of a vector are positive.

3 Solving Linear Systems $Ax = b$

Open-Ended Questions

1. What is the difference between a direct method and an iterative method?
2. When is Gaussian elimination preferred over Gauss–Seidel?
3. Why does matrix structure (dense vs. tridiagonal) matter computationally?

Hand Calculations

1. Solve the system using Gaussian elimination:

$$\begin{aligned} 2x_1 - x_2 &= 1, \\ -x_1 + 2x_2 - x_3 &= 0, \\ -x_2 + 2x_3 &= 1. \end{aligned}$$

2. Perform *one full Gauss–Seidel sweep* starting from $x^{(0)} = (0,0,0)^T$ for the system:

$$\begin{aligned} 10x_1 + 2x_2 + x_3 &= 13, \\ 2x_1 + 10x_2 + 3x_3 &= 14, \\ x_1 + 3x_2 + 10x_3 &= 15. \end{aligned}$$

Pseudocode

1. Write pseudocode for the Thomas algorithm (forward elimination and back substitution).

2. Write pseudocode for one Gauss–Seidel iteration for a general $n \times n$ system.

4 Nonlinear Equations and Newton's Method

Open-Ended Questions

1. What does it mean to linearize a nonlinear equation?

2. Why is Newton's method considered a root-finding algorithm?

3. What conditions are required for quadratic convergence?

4. Why does Newton's method require solving a linear system at each iteration (for systems)?

Hand Calculations

1. Derive Newton's update formula for solving $f(x) = 0$ using a Taylor series expansion.

2. Given $f(x) = x^2 - 2$, perform two Newton iterations starting from $x_0 = 1$.

3. For the nonlinear function

$$f(\sigma) = \frac{\sigma}{E} + \alpha \left(\frac{\sigma}{\sigma_0} \right)^m - \varepsilon,$$

identify $f'(\sigma)$ symbolically.

Pseudocode

1. Write pseudocode for Newton's method for a scalar equation.

2. Write pseudocode for Newton's method applied to a system of nonlinear equations using a Jacobian matrix.

5 Big Picture Questions

1. Why do so many engineering problems reduce to solving $Ax = b$ or $F(x) = 0$?
2. How does Newton's method for nonlinear systems mirror what we do in computational mechanics solvers?
3. Explain how symbolic computation (SymPy) and numerical computation (NumPy) complement each other.

Study Advice: Be prepared to explain ideas in words, carry out small calculations cleanly by hand, and write clear, logically structured pseudocode. The exam emphasizes understanding over memorization.