Mock Exam 2025

EPFL	**Course:**	Numerical Analysis – EL/MX/CH
	Lecturer:	Prof. Michael Herbst
	Date:	Spring Session 2025 **Duration:** 3h 00

Mock exam disclaimer

- This mock exam gives you an example how the final exam will look like. Questions 1–4, 7, & 8 have been taken from the **2024 exam**, question 5 & 6 from the **2024 mock exam**. As the course has changed compared to last year, some questions may use terminology we did not employ this year.
- In 2025, the final exam will only contain **pen and paper** questions, which moreover will cover the contents of **the entire class**. Furthermore you can expect this year's **final exam** to be a little **more involved** than this mock exam.

Exam instructions

This exam has **8 exercises** with in total **45 points**. The material consists of **two** parts:

- This **question sheet** with the questions. All questions are pen and paper questions.
- A personalised **answer sheet** (with your name and sciper number).

Answering the questions

- For each question on this question sheet, provide the **answers in the corresponding section** of the **answer sheet**. **Do not write onto the question sheet itself**. Only these answer sheets will be marked.
- On the **answer sheet** only **write within the black boxes**. If you need extra space, additional blank pages are given in the back. **Clearly identify for which question** you provide additional answers. **Also add a remark in the original answer box** where you run out of space that additional text can be found in the appendix.
- **Please write with a pen (no pencil or erasable ballpen)**.
- For each of your answers, **outline the reasoning** and **justify your answer**.
- At the end of the exam both the question and answer sheet will be collected.

Authorised material

- You are allowed **a two-sided handwritten A4 cheatsheet** (hand-written on paper, no print-outs).
- No other notes, sheets or books are allowed. No calculator, mobile phone, tablet, laptop or any other electronic device.

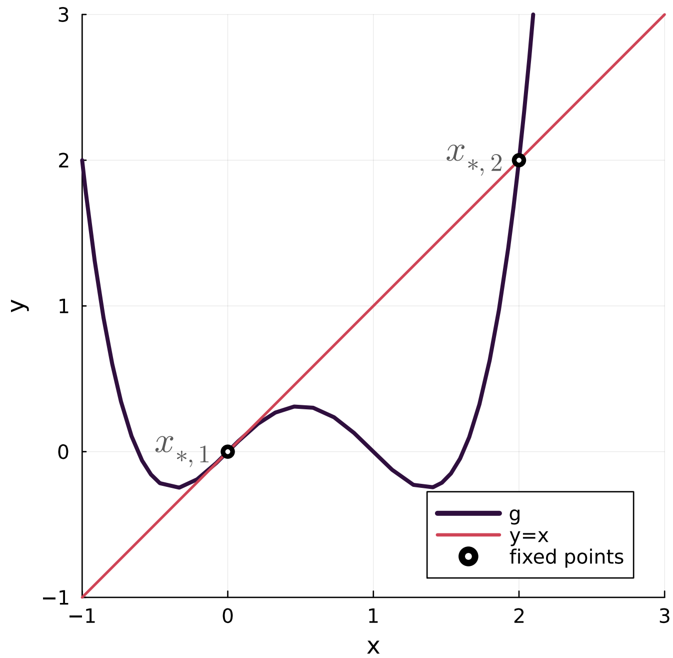
Exercise 1 (7P) — exam 2024

Given a parameter $\theta \in \mathbb{R}$, consider the function

$$g(x) = x^4 - 2\theta x^3 + x.$$

(a) **(2P)** Show that the only fixed points of g are $x_{*,1} = 0$ and $x_{*,2} = 2\theta$.

(b) **(1P)** Figure 1 depicts $g(x)$ for $\theta = 1$ together with its two fixed points $x_{*,1}$, $x_{*,2}$. By visual inspection determine for which of the two fixed points $x_{*,1}$ and $x_{*,2}$ the fixed-point iterations $x^{(k+1)} = g(x^{(k)})$ converge, provided that a starting point $x^{(0)}$ sufficiently close to the respective fixed point has been chosen. Justify your answer.



(c) **(3P)** We return to the general case where θ is a parameter of the problem.

(i) For which values of θ do fixed-point iterations converge to $x_{*,2}$ provided a good starting point is chosen?

(ii) For which values of θ is the fastest convergence rate to $x_{*,2}$ achieved ?

(d) We consider the case where θ is chosen such that the fastest convergence rate in the fixed-point iterations is achieved (the value you determined in (c) (ii)).

(1P) What is the convergence order of Newton's method for these value(s) of θ? Does Newton provide any advantage over fixed point iterations in this case?

Exercise 2 (8P) — exam 2024

(a) **(1.5P)** We are given $n+1$ nodes $x_1, \dots, x_{n+1} \in \mathbb{R}$. Define the Lagrange basis associated to $x_1, \dots, x_{n+1}$ and specify how this basis can be used to find the n-th degree interpolating polynomial through the data points $(x_1, y_1), (x_2, y_2), \dots (x_{n+1}, y_{n+1})$.

(b) **(1.5P)** Using the Lagrange basis, find the interpolating polynomial through the points $(x_1 = -1, y_1 = -2)$, $(x_2 = 1, y_2 = 0)$, $(x_3 = 4, y_3 = 6)$.

For polynomial interpolation with equally spaced nodes, the interpolation error is governed by the following theorem:

Theorem. For a C^{n+1} function $f : [a,b] \to \mathbb{R}$ and $a = x_1 < x_2 < \cdots < x_{n+1} = b$ equally distributed nodes in $[a,b]$ the n-th degree polynomial interpolant p_n of the data $(x_i, f(x_i))$ with $i = 1, 2, \dots, n+1$ satisfies the following bound on the interpolation error

$$\max_{x \in [a,b]} |f(x) - p_n(x)| \leq \frac{1}{4(n+1)} \left(\frac{b-a}{n}\right)^{n+1} \max_{x \in [a,b]} \left| f^{(n+1)}(x) \right|. \tag{1}$$

We consider polynomial interpolation with $n+1$ equally distributed nodes over the interval $[-1,1]$ for the functions $f_1(x) = \sin(x)$ and $f_2(x) = \frac{1}{1+20x^2}$.

(c) **(3P)** Show that for f_1 the interpolation error goes to 0 as $n \to \infty$.

(d) **(2P)** For f_2 we have $\max_{x \in [0,1]} \left| f_2^{(n+1)}(x) \right| \approx 20^n n!$ such that the right-hand side of (1) grows to infinity as $n \to \infty$. What happens to the polynomial interpolation in this case? What needs to be changed in the polynomial interpolation procedure to achieve exponential convergence for such functions f_2?

Exercise 3 (7P) — exam 2024

We consider the matrix

$$\mathbf{A} = \begin{pmatrix} 1 & \alpha \\ \alpha & 1 \end{pmatrix}$$

for $\alpha > 0$ and an associated linear system

$$\mathbf{A}\mathbf{x} = \mathbf{b} \quad \text{with} \quad \mathbf{b} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \tag{2}$$

where we seek the solution $\mathbf{x} \in \mathbb{R}^2$.

When representing (2) on a computer we assume that the available floating-point precision is unable to represent $\mathbf{b}$ exactly introducing a small error $\varepsilon > 0$: the computer is only able to solve the approximate linear system

$$\mathbf{A}\widetilde{\mathbf{x}} = \widetilde{\mathbf{b}} \quad \text{with} \quad \widetilde{\mathbf{b}} = \begin{pmatrix} 1+\varepsilon \\ 1-\varepsilon \end{pmatrix} \tag{3}$$

and thus only able to obtain an approximate solution $\widetilde{\mathbf{x}} \in \mathbb{R}^2$.

(a) **(2P)** For a *general* square and invertible matrix $\mathbf{M} \in \mathbb{R}^{n\times n}$ define the condition number $\kappa(\mathbf{M})$ in terms of matrix norms. Also provide an expression to compute the condition number using eigenvalues of appropriate matrices.

(b) **(2P)** Show that the condition number of $\mathbf{A}$ is

$$\kappa(\mathbf{A}) = \left| \frac{1+\alpha}{1-\alpha} \right|. \tag{4}$$

You may use that a matrix $\left(\begin{smallmatrix} a & c \\ c & b \end{smallmatrix}\right) \in \mathbb{R}^{2\times 2}$ has eigenvalues

$$\frac{a+b}{2} - \sqrt{\frac{(a-b)^2}{4} + c^2} \qquad \text{and} \qquad \frac{a+b}{2} + \sqrt{\frac{(a-b)^2}{4} + c^2}.$$

(c) **(1P)** The solution to the perturbed system (3) is given by

$$\widetilde{\mathbf{x}} = \frac{1}{1+\alpha}\begin{pmatrix} 1 \\ 1 \end{pmatrix} + \frac{\varepsilon}{1-\alpha}\begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

and similarly $\mathbf{x} = \frac{1}{1+\alpha}\left(\begin{smallmatrix} 1 \\ 1 \end{smallmatrix}\right)$. Compute the relative error in the solution $\frac{\|\widetilde{\mathbf{x}}-\mathbf{x}\|}{\|\mathbf{x}\|}$ as well as the relative error in the right-hand side $\frac{\|\widetilde{\mathbf{b}}-\mathbf{b}\|}{\|\mathbf{b}\|}$.

(d) **(2P)** Describe in one sentence what the condition number measures for a linear system (2). Use this to explain your results in (c).

Exercise 4 (5P) — exam 2024

Consider the algorithm for **LU factorization** given below.

Algorithm (LU factorisation).
Input: $\mathbf{A} \in \mathbb{R}^{n \times n}$,
Output: $\mathbf{U} \in \mathbb{R}^{n \times n}$, $\mathbf{L} \in \mathbb{R}^{n \times n}$

1. $\mathbf{A}^{(1)} = \mathbf{A}$
2. for $k = 1, \ldots, n-1$ *(algorithm steps)*
    1. $L_{kk} = 1$
    2. for $i = k+1, \ldots, n$ *(Loop over rows)*
        1. $L_{ik} = \frac{A_{ik}^{(k)}}{A_{kk}^{(k)}}$
        2. for $j = k+1, \ldots n$ *(Loop over columns)*
            1. $A_{ij}^{(k+1)} = A_{ij}^{(k)} - L_{ik} A_{kj}^{(k)}$
3. $\mathbf{U} = \mathbf{A}^{(n)}$

(a) **(1.5P)** Making reference to the algorithm explain why LU factorisation is said to have a computational cost of $\mathcal{O}(n^3)$.

(b) **(1.5P)** Let $\mathbf{A} \in \mathbb{R}^{n \times n}$ be an n by n square matrix. If $\mathbf{A}$ has no special structure what is the memory usage for storing $\mathbf{A}$? What is the memory usage for storing the $\mathbf{L}$ and $\mathbf{U}$ factors once LU factorisation has been performed? Specify your answer using big $\mathcal{O}$ notation.

(c) Now assume that $\mathbf{A}$ is sparse.

- **(1P)** If we only store the non-zero elements of the matrix explicitly, what is the memory cost of $\mathbf{A}$ in this case?
- **(1P)** Explain the phaenomenon called fill-in and its consequences for the memory cost of the $\mathbf{L}$ and $\mathbf{U}$ factors of sparse matrices.

Exercise 5 (7P) — mock 2024

Let $\mathbf{A}\mathbf{x} = \mathbf{b}$ be a linear system with given $\mathbf{A} \in \mathbb{R}^{n\times n}$ and $\mathbf{b} \in \mathbb{R}^n$. We consider the fixed-point map

$$g(\mathbf{x}) = \mathbf{x} + \mathbf{P}^{-1}(\mathbf{b} - \mathbf{A}\mathbf{x}) \tag{5}$$

for a invertible preconditioner matrix $\mathbf{P} \in \mathbb{R}^{n\times n}$. The iterative procedure $\mathbf{x}^{(k+1)} = g(\mathbf{x}^{(k)})$ starting from an initial vector $\mathbf{x}^{(0)} \in \mathbb{R}^n$ is the Richardson iteration.

(a) **(1P)** Show that if $\mathbf{x} \in \mathbb{R}^n$ is a fixed point of g than it is also a solution to the linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$.

(b) **(2P)** Show that if $\mathbf{x}^{(k+1)} = g(\mathbf{x}^{(k)})$ and if $\mathbf{x}$ is a fixed point of g, then

$$(\mathbf{x}^{(k+1)} - \mathbf{x}) = (\mathbf{I} - \mathbf{P}^{-1}\mathbf{A})(\mathbf{x}^{(k)} - \mathbf{x}). \tag{6}$$

(c) **(1.5P)** Based on (6) give the conditions for Richardson iterations $\mathbf{x}^{(k+1)} = g(\mathbf{x}^{(k)})$ to converge to a fixed point $\mathbf{x}$ independent of the chosen initial vector $\mathbf{x}^{(0)}$ and right-hand side $\mathbf{b}$.

Consider the case

$$\mathbf{A} = \begin{pmatrix} 2 & 0 \\ 0 & \alpha \end{pmatrix} \qquad \mathbf{P} = \begin{pmatrix} \beta & 0 \\ 0 & 1 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \qquad \mathbf{x}^{(0)} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

where $\alpha \in \mathbb{R}$ and $\beta \in \mathbb{R}$ and consider the fixed-point iterations $\mathbf{x}^{(k+1)} = g(\mathbf{x}^{(k)})$ for $k = 0, 1, \ldots$.

(d) **(1.5P)** For which choice of α and β do the fixed-point iterations converge.

(e) **(1P)** For which choice of α and β do the fixed-point iterations converge at the fastest possible rate. How may iteration steps are at most needed for convergence ?

Exercise 6 (5P) — mock 2024

Consider $\mathbf{b} \in \mathbb{R}^2$ and the following 2×2 matrix

$$\mathbf{A} = \begin{pmatrix} 1 & -\beta \\ -\frac{1}{2} & 1 \end{pmatrix},$$

for a real number $\beta \in \mathbb{R}$. Consider Jacobi's method to solve the linear system $\mathbf{Ax} = \mathbf{b}$.

(a) **(2.5P)** State the conditions for Jacobi's method to converge for any right-hand side $\mathbf{b}$ and initial vector $\mathbf{x}^{(\mathbf{0})}$.

(b) **(2.5P)** Deduce conditions for the parameter β, which ensure convergence for any right-hand side $\mathbf{b}$ and initial vector $\mathbf{x}^{(\mathbf{0})}$.

Exercise 7 (6P) — exam 2024

We consider the family of triangular matrices

$$\mathbf{A} = \begin{pmatrix} \lambda_1 & 1 & 1 \\ 0 & \lambda_2 & 1 \\ 0 & 0 & \lambda_3 \end{pmatrix}$$

where $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq 0$.

(a) **(3P)** We consider the power iterations on the matrix $\mathbf{A}$ starting from an initial guess $\mathbf{x}^{(0)} = \left(\begin{smallmatrix}1\\1\\1\end{smallmatrix}\right)$. State conditions on the parameters λ_1, λ_2 and λ_3 for the power iterations to converge. To which eigenvalue will they converge? Specify the convergence order and provide the convergence rate in terms of λ_1, λ_2 and λ_3.

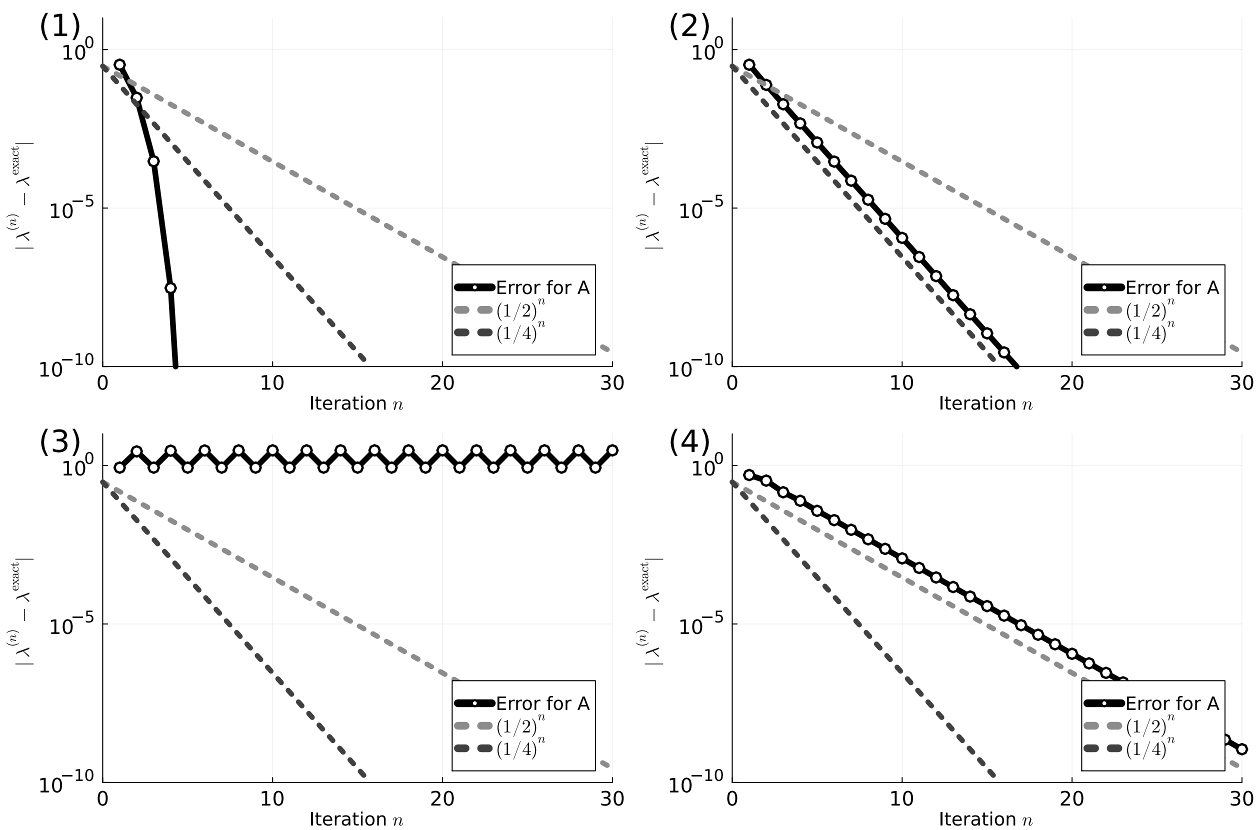
(b) **(1P)** Prove for a general matrix $\mathbf{M} \in \mathbb{R}^{n\times n}$:

> If $(\alpha, \mathbf{x})$ is an eigenpair of $\mathbf{M}$, i.e. $\mathbf{M}\mathbf{x} = \alpha\mathbf{x}$, and $\mathbf{M}$ is invertible, then $\left(\frac{1}{\alpha}, \mathbf{x}\right)$ is an eigenpair of $\mathbf{M}^{-1}$.

(c) **(2P)** Consider the matrix

$$\mathbf{B} = \begin{pmatrix} 5 & 1 & 1 \\ 0 & 2 & 1 \\ 0 & 0 & 1 \end{pmatrix},$$

which is a special case of $\mathbf{A}$ for $\lambda_1 = 5$, $\lambda_2 = 2$, $\lambda_3 = 1$. We perform *shifted inverse iterations* with shift $\sigma = 4$. Which eigenvalue λ_{exact} is targeted? Which of the following four plots is obtained? Justify your choice making reference to the discussion in the previous parts of the exercise.



Exercise 8 (4P) — exam 2024

Let $f : [a, b] \to \mathbb{R}$ be a real-valued function with $0 < a < b$. We consider a numerical integration formula

$$Q(f) = h \sum_{i=0}^{n} w_i f(t_i)$$

with $n + 1$ equispaced quadrature nodes

$$t_i = a + ih \quad \text{for } i = 0, \dots, n \qquad \text{and} \qquad h = \frac{b - a}{n}$$

as well as weights w_i for $i = 0, \dots, n$. $Q(f)$ approximates $\int_a^b f(x)\,\mathrm{d}x$.

(a) **(0.5P)** Define the **degree of exactness** of Q.

(b) **(1P)** State the trapezoid formula for computing $\int_a^b f(x)\mathrm{d}x$ and provide its degree of exactness.

(c) **(2.5P)** In the lecture we discussed

> **Theorem.** If a numerical integration formula Q has a degree of exactness r then the formula is of order $r + 1$, i.e.
>
> $$\left| \int_a^b f(x)\,dx - Q(f) \right| \leq C\,h^{r+1}$$
>
> where C is a constant independent of h.

Inspect the following convergence graphs and apply this theorem to determine the degree of exactness of the two quadrature formulae (**Formula A** and **Formula B**). Which of the two formulae behaves like the trapezoid method?