UNIVERSITY OF OSLO

Faculty of mathematics and natural sciences

Exam in:	MAT3110/MAT4110 — Introduction to numerical analysis
Day of examination:	17 December 2025
Examination hours:	0900 – 1300
This problem set consists of 6 pages.	
Appendices:	None
Permitted aids:	None

Please make sure that your copy of the problem set is complete before you attempt to answer anything.

All 10 part questions will be weighted equally.

Problem 1 Lower triangular matrix

Let A be a non-singular lower triangular $n \times n$ matrix.

1a

Write a Matlab (or python) routine `function x=forwardsolve(A,b)` that returns the solution $\mathbf{x} \in \mathbb{R}^n$ to the linear system $A\mathbf{x} = \mathbf{b}$, given $\mathbf{b} \in \mathbb{R}^n$.

Answer: The i-th equation is

$$\sum_{j=1}^{i} a_{ij} x_j = b_i,$$

and so

$$x_i = \Big(b_i - \sum_{j=1}^{i-1} a_{ij} x_j \Big) \Big/ a_{ii}.$$

Here is the code:

```
function x=forwardsolve(A,b)
n=length(b); x=b;
x(1)=b(1)/A(1,1);
for i=2:n
  s = A(i,1) * x(1);
  for j=2:i-1
    s = s + A(i,j) * x(j);
  end
```

(Continued on page 2.)

```
  x(i)=(b(i)-s)/A(i,i);
end
```

1b

How many flops are required to compute $\mathbf{x}$?

Answer: To compute x_1 requires one division. To compute x_i for $i \geq 2$ requires $i-1$ multiplications, $i-2$ additions, one subtraction and one division, a total of $2i-1$ flops. So the number of flops required to compute $\mathbf{x}$ is

$$\sum_{i=1}^{n}(2i-1) = 2\Big(\sum_{i=1}^{n} i\Big) - n = 2\frac{n(n+1)}{2} - n = n^2.$$

Problem 2 Cholesky

Find the Cholesky factorization, LDL^T, of the matrix

$$A = \begin{bmatrix} 2 & -4 \\ -4 & 11 \end{bmatrix}.$$

Is A positive definite?

Answer We initialize $A_0 = A$. In the first step use the first column of A_0:

$$\mathbf{l}_1 = \frac{1}{2}\begin{bmatrix} 2 \\ -4 \end{bmatrix} = \begin{bmatrix} 1 \\ -2 \end{bmatrix},$$

and the first diagonal element of A_0, $D_{1,1} = 2$. Then

$$\mathbf{l}_1\mathbf{l}_1^T = \begin{bmatrix} 1 & -2 \\ -2 & 4 \end{bmatrix},$$

and

$$D_{1,1}\mathbf{l}_1\mathbf{l}_1^T = \begin{bmatrix} 2 & -4 \\ -4 & 8 \end{bmatrix},$$

and

$$A_1 = A_0 - D_{1,1}\mathbf{l}_1\mathbf{l}_1^T = \begin{bmatrix} 0 & 0 \\ 0 & 3 \end{bmatrix}.$$

In the second step we set

$$\mathbf{l}_2 = \frac{1}{3}\begin{bmatrix} 0 \\ 3 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

and $D_{2,2} = 3$. So

$$L = \begin{bmatrix} 1 & 0 \\ -2 & 1 \end{bmatrix}, \qquad D = \begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix}.$$

By a theorem from the course, since A is symmetric, A is positive definite if and only if it has an LDL^T factorization in which the diagonal elements of D are positive. Since $D_{11}, D_{22} > 0$, A is positive definite.

(Continued on page 3.)

Problem 3 Iterative scheme

Consider the linear system $A\mathbf{x} = \mathbf{b}$ where A is a non-singular $n \times n$ matrix and $\mathbf{b}$ is a vector in $\mathbb{R}^n$. An iterative scheme for solving the system can be derived from a splitting matrix B. What is then the iterative scheme and what is the condition on A and B to guarantee convergence?

Answer: We subtract $B\mathbf{x}$ from each side so that

$$(A - B)\mathbf{x} = -B\mathbf{x} + \mathbf{b}.$$

The associated scheme is then

$$(A - B)\mathbf{x}^{(k+1)} = -B\mathbf{x}^{(k)} + \mathbf{b}, \quad k = 0, 1, 2, \ldots,$$

with $\mathbf{x}^{(0)}$ the initial guess.

Let the error in $\mathbf{x}^{(k)}$ be $\mathbf{e}^{(k)} := \mathbf{x}^{(k)} - \mathbf{x}$. Then

$$(A - B)\mathbf{e}^{(k+1)} = -B\mathbf{e}^{(k)},$$

and so

$$\mathbf{e}^{(k+1)} = H\mathbf{e}^{(k)},$$

where

$$H := -(A - B)^{-1}B.$$

By a theorem from the notes, the error vectors $\mathbf{e}^{(k)}$ converge to zero as $k \to \infty$ if and only if $\rho(H) < 1$, where $\rho(H)$ is the spectral radius,

$$\rho(H) := \max_{i=1,\ldots,n} |\lambda_i|,$$

and the λ_i are the eigenvalues of H.

Problem 4 Polynomial interpolation

The Chebyshev polynomial of degree $n \geq 0$ is

$$T_n(x) := \cos(n \arccos x), \qquad -1 \leq x \leq 1.$$

4a

Derive a formula for T_{n+1} as a function of T_n and T_{n-1}, and use it to find T_2.

Answer: Let $n \geq 1$ and let $\theta := \arccos x$. By the angle sum identity for the cosine function,

$$T_{n+1}(x) = \cos(n\theta + \theta) = \cos(n\theta)\cos(\theta) - \sin(n\theta)\cos(\theta),$$
$$T_{n-1}(x) = \cos(n\theta - \theta) = \cos(n\theta)\cos(\theta) + \sin(n\theta)\cos(\theta).$$

So,

$$T_{n+1}(x) + T_{n-1}(x) = 2\cos(n\theta)\cos(\theta) = 2xT_n(x),$$

and

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x).$$

Letting $n = 1$ gives

$$T_2(x) = 2x^2 - 1.$$

(Continued on page 4.)

4b

What are the zeros of T_2 and why are they suitable for polynomial interpolation?

Answer: The zeros of T_2 are $\pm 1/\sqrt{2}$. If we interpolate a function f by a linear polynomial p at points x_0 and x_1 in $[-1, 1]$, then the error at $x \in [-1, 1]$ is given by

$$e(x) = \frac{f''(\xi)}{2!}\omega(x),$$

for some $\xi \in [-1, 1]$, where

$$\omega(x) := (x - x_0)(x - x_1).$$

The maximum norm of ω in $[-1, 1]$ is minimized by choosing the x_i to be the zeros of T_2, i.e., $x_i = \pm 1/\sqrt{2}$. Thus we can expect a small interpolation error with this choice.

Problem 5 Bernstein basis

A Bézier curve of degree d in $\mathbb{R}^2$ is a parametric curve

$$\mathbf{p}(t) := \sum_{i=0}^{d} B_{i,d}(t)\mathbf{c}_i, \qquad 0 \le t \le 1,$$

with control points $\mathbf{c}_0, \mathbf{c}_1, \ldots, \mathbf{c}_d \in \mathbb{R}^2$, and

$$B_{i,d}(t) := \binom{d}{i} t^i (1 - t)^{d-i}.$$

Show that the length of $\mathbf{p}$ is bounded above by the length of its control polygon (the polygon passing through its control points). Hint: you can use the fact that

$$\mathbf{p}'(t) = d \sum_{i=0}^{d-1} B_{i,d-1}(t)(\mathbf{c}_{i+1} - \mathbf{c}_i).$$

Answer: The length of $\mathbf{p}$ is

$$L(\mathbf{p}) = \int_0^1 \|\mathbf{p}'(t)\|\, dt,$$

where $\|\cdot\|$ is the Euclidean norm in $\mathbb{R}^2$. By the formula for $\mathbf{p}'$,

$$\begin{aligned}\|\mathbf{p}'(t)\| &= d\Big\| \sum_{i=0}^{d-1} B_{i,d-1}(t)(\mathbf{c}_{i+1} - \mathbf{c}_i)\Big\| \\ &\le d \sum_{i=0}^{d-1} |B_{i,d-1}(t)|\|\mathbf{c}_{i+1} - \mathbf{c}_i\|,\end{aligned}$$

(Continued on page 5.)

by the triangle inequality for the Euclidean norm. So

$$L(\mathbf{p}) \le d \sum_{i=0}^{d-1} \int_0^1 |B_{i,d-1}(t)|\, dt \|\mathbf{c}_{i+1} - \mathbf{c}_i\|.$$

Since $B_{i,d-1}(t) \ge 0$ for $0 \le t \le 1$, and since

$$\int_0^1 B_{i,d-1}(t)\, dt = 1/d$$

(by a theorem from the course notes), we conclude that

$$L(\mathbf{p}) \le \sum_{i=0}^{d-1} \|\mathbf{c}_{i+1} - \mathbf{c}_i\|.$$

Problem 6 Newton's method

6a

Given $f : \mathbb{R} \to \mathbb{R}$, what is Newton's method for finding a root x_* of the equation $f(x_*) = 0$?

Answer: Newton's method is

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}, \qquad k = 0, 1, 2, \ldots,$$

where x_0 is an initial guess.

6b

Let

$$f(x) := x^2(x+1), \qquad x \in \mathbb{R}.$$

What is the expected order of convergence to the root $x_* = -1$? Determine the order of convergence to the root $x_* = 0$.

Answer: Let $e_k := x_k - x_*$. From the course notes,

$$e_{k+1} \approx C e_k^2$$

as $x_k \to x_*$ as long as $f'(x_*) \ne 0$. Since

$$f'(x) = 3x^2 + 2x,$$

$f'(-1) = 1 \ne 0$ and so the convergence to $x_* = -1$ is quadratic.

For the convergence to $x_* = 0$, Newton's method is

$$x_{k+1} = x_k - \frac{x_k^2(x_k+1)}{3x_k^2 + 2x_k} = x_k \frac{2x_k + 1}{3x_k + 2} \approx \frac{1}{2} x_k$$

for x_k close to 0. Thus the method finds the root $x_* = 0$ with linear convergence.

(Continued on page 6.)

Problem 7 Gradient descent

Suppose A is a symmetric positive definite $n \times n$ matrix and $\mathbf{b} \in \mathbb{R}^n$. The gradient descent method attempts to solve $A\mathbf{x}^* = \mathbf{b}$ by iteratively minimizing

$$F(\mathbf{x}) := \frac{1}{2}\mathbf{x}^T A\mathbf{x} - \mathbf{x}^T\mathbf{b}, \qquad \mathbf{x} \in \mathbb{R}^n.$$

If $\mathbf{x}$ is the current approximation to $\mathbf{x}^*$, we choose a direction $\mathbf{d} \in \mathbb{R}^n$ such that $\mathbf{d}^T\nabla F(\mathbf{x}) < 0$ and the new approximation is

$$\mathbf{x}' = \mathbf{x} - \frac{\mathbf{d}^T\nabla F(\mathbf{x})}{\mathbf{d}^T A\mathbf{d}}\mathbf{d}.$$

Show that the new gradient, $\nabla F(\mathbf{x}')$, is orthogonal to $\mathbf{d}$.

Answer: The gradient of F is

$$\nabla F(\mathbf{x}) = A\mathbf{x} - \mathbf{b}.$$

Therefore

$$\nabla F(\mathbf{x}') = A\mathbf{x}' - \mathbf{b} = A\mathbf{x} - \frac{\mathbf{d}^T\nabla F(\mathbf{x})}{\mathbf{d}^T A\mathbf{d}}A\mathbf{d} - \mathbf{b},$$

and so

$$\begin{aligned}
\mathbf{d}^T\nabla F(\mathbf{x}') &= \mathbf{d}^T A\mathbf{x} - \frac{\mathbf{d}^T\nabla F(\mathbf{x})}{\mathbf{d}^T A\mathbf{d}}\mathbf{d}^T A\mathbf{d} - \mathbf{d}^T\mathbf{b} \\
&= \mathbf{d}^T A\mathbf{x} - \mathbf{d}^T\nabla F(\mathbf{x}) - \mathbf{d}^T\mathbf{b} \\
&= \mathbf{d}^T(A\mathbf{x} - \nabla F(\mathbf{x}) - \mathbf{b}) \\
&= \mathbf{d}^T\mathbf{0} = 0.
\end{aligned}$$

Good luck!