Answer sheet compilation instructions

- Use only black or blue pen.

For open answers:
- Write **clearly** only inside the provided boxes.
- Each box should contain a single integer (positive or negative) or a single fraction (reduced to lowest form).

For multiple choice questions:
- Fill the circle of the answer you consider correct (only one answer is correct).
- Remarks and computations have **no** influence on points awarded.
- Any unclear or double marks will be counted as answer not given (0 points).
- Wrong answers give **negative** points.

Exam instructions

- Turn off your devices and leave them in your bag.
- Only pens and Legi should be on the table.
- Fill your last name and Legi number on the answer sheet.
- Turn this sheet only when instructed to do so.
- At the end of the exam, give the single answer sheet which you want to submit to an assistant, and take everything else with you.
1. *Fourier transform* [5 points].

(a) Which of the functions in the figures

Figure 1:  

Figure 2:  

Figure 3:  

has the following Fourier transform (absolute values of the Fourier coefficients are plotted):

Solution: Figure 2 [1,0,-1]

(b) Consider the vectors 

\[ a := (1, 1, 2, 0) \quad \text{and} \quad b := (2, 1, 3, 2), \]

their 4-periodic convolution \( c := a *_4 b \) and the Fourier matrix

\[
\mathbf{F}_4 = \begin{pmatrix}
1 & 1 & 1 & 1 \\
1 & -i & -1 & i \\
1 & -1 & 1 & -1 \\
1 & i & -1 & -i
\end{pmatrix}.
\]
Compute the discrete Fourier transform of \( c \).

\[ \text{Solution: } \text{DFT}(c) = (\ldots, 32, 2, 4, 2, \ldots) \] [2 if all correct and 0 otherwise.]

(c) Consider the C++/Eigen code

```cpp
c. Eigen::VectorXcd dft_1(const Eigen::VectorXcd &y) {
    const double pi = 3.14159265359;
    const int n = y.size();
    const int n_half = n / 2;
    if (n == 1) return y;
    Eigen::VectorXcd y1(n_half);
    Eigen::VectorXcd y2(n_half);
    for (int i = 0; i < n_half; ++i) {
        y1(i) = y(2 * i);
        y2(i) = y(2 * i + 1);
    }
    const Eigen::VectorXcd c1 = dft_1(y1);
    const Eigen::VectorXcd c2 = dft_1(y2);
    const std::complex<double> omega = std::exp(-2.0 * pi / n * std::complex<double>(0.0, 1.0));
    std::complex<double> omega_k(1.0, 0.0);
    Eigen::VectorXcd c(n);
    for (int k = 0; k < n; ++k) {
        c(k) = c1(k % n_half) + c2(k % n_half) * omega_k;
        omega_k *= omega;
    }
    return c;
}

c. Eigen::VectorXcd dft_2(const Eigen::VectorXcd &y) {
    const double pi = 3.14159265359;
    const int n = y.size();
    Eigen::MatrixXcd F(n, n);
    for (int j = 0; j < n; ++j) {
        for (int i = 0; i < n; ++i) {
            F(i, j) = std::exp(-2.0 * pi / n * i * j * std::complex<double>(0.0, 1.0));
        }
    }
    return F * y;
}
```

You may assume that the length \( n \geq 2 \) of the vector \( y \) is a power of 2. The two functions \texttt{dft}_1 and \texttt{dft}_2 perform the same task, but with a different algorithm. Provide the lowest numbers \( p, q, \alpha, \beta \in \mathbb{N}_0 \) such that the asymptotic complexities for large \( n \) are given by:

- \( \texttt{dft}_1: \mathcal{O}(n^p \log^q(n)) \)
- \( \texttt{dft}_2: \mathcal{O}(n^\alpha \log^\beta(n)) \)

\[ \text{Solution: } p = q = 1 \text{ and } \alpha = 2, \beta = 0. \] [+1 for every correct pair.]
2. Chebyshev interpolation [3 points].

Let $L_T f$ denote the polynomial interpolant of the function $f : I \rightarrow \mathbb{R}$ for the set of Chebyshev nodes $T := \{t_0, t_1, \ldots, t_n\}$, $n \in \mathbb{N}$. What kind of convergence, with respect to $n$, is to be expected for the approximation error of the Chebyshev interpolant ($\|f - L_T f\|_{L^\infty(I)}$) in each of the following cases?

(a) $f \in C^\infty(I)$

**Solution:** Exponential convergence

Points: $[+1,0,-1]$

(b) $f \in C^2(I)$, but $f \not\in C^3(I)$

**Solution:** Algebraic convergence

Points: $[+1,0,-1]$

(c) $f \in C^0(I)$, but $f \not\in C^1(I)$

**Solution:** Algebraic convergence

Points: $[+1,0,-1]$

3. Convergence of Gauss-Legendre Quadrature Formula [3 points].

Let $Q_n(f)$ be the $n$-point Gauss-Legendre quadrature rule on $\Omega \subset \mathbb{R}$ for an integrand $f$. What kind of convergence, with respect to $n$, is to be expected for the quadrature error $E_n(f) := \left| \int_\Omega f(t) \, dt - Q_n(f) \right|$ in each of the following cases:

(a) $f(t) = t^{5/2}$ and $\Omega = [0, 1]$

**Solution:** Algebraic convergence

Points: $[+1,0,-1]$

For $f \in C^r(\Omega)$, Gauss-Legendre quadrature formula converges algebraically with $O(n^{-r})$. Here $f \in C^2(\Omega)$.

(b) $f \in C^\infty(\Omega)$

**Solution:** Exponential convergence

Points: $[+1,0,-1]$

For a smooth function, Gauss-Legendre quadrature formula converges exponentially with $O(\lambda^n)$, $\lambda \in (0, 1)$.

(c) $f(t) = |t|$ and $\Omega = [-1, 1]$

**Solution:** Algebraic convergence

Points: $[+1,0,-1]$

This is because $f \in C^0(\Omega)$.

4. Convergence of Composite Simpson rule [3 points].

Let $Q_n(f)$ be the $n$-point Composite Simpson quadrature rule with equally spaced nodes on $\Omega \subset \mathbb{R}$ for an integrand $f$. What kind of convergence, with respect to $n$, is to be expected for the quadrature error $E_n(f) := \left| \int_\Omega f(t) \, dt - Q_n(f) \right|$ in each of the following cases:

(a) $f(t) = t^{5/2}$ and $\Omega = [0, 1]$

**Solution:** Algebraic convergence

Points: $[+1,0,-1]$

For $f \in C^r(\Omega)$, Composite quadrature formula (with local order $q$) converges algebraically with $O(n^{-\min\{r,q\}})$. Here $f \in C^2(\Omega)$.

(b) $f(t) = t^{5/2}$ and $\Omega = [1, 2]$

**Solution:** Algebraic convergence

Points: $[+1,0,-1]$

Here $f \in C^\infty(\Omega)$. Regardless, Composite quadrature formula (with local order $q$) converges algebraically with $O(n^{-q})$.

(c) $f(t) = |t|$ and $\Omega = [-1, 1]$

**Solution:** Algebraic convergence

Points: $[+1,0,-1]$

This is because $f \in C^0(\Omega)$.
5. Runge-Kutta methods [5 points].

(a) Consider the following 2-stage Runge-Kutta method used for solving the ODE \( \dot{y} = f(y) \):

\[ y_{k+1} = y_k - \frac{h}{2} (k_1 - 3k_2) \]

where

\[ k_1 = f \left( t_k + \frac{h}{2}, y_k + \frac{h}{2} k_1 \right), \]
\[ k_2 = f \left( t_k + \frac{3h}{2}, y_k - \frac{h}{2} (k_1 - 4hk_2) \right). \]

Complete the entries of the Butcher tableau corresponding to this method.

Solution: (+2 if all entries are correct, no negatives)

There was a typo in this subproblem. The term marked in red should have been \(-4k_2\) instead of \(-4hk_2\).

The correct Butcher tableau is:

| \( \frac{1}{2} \) | \( \frac{1}{2} \) | 0 |
| \( \frac{3}{2} \) | \( -\frac{1}{2} \) | 2 |
| \( -\frac{1}{2} \) | \( \frac{3}{2} \) |

The entry in blue is the one affected by the typo. During correction of this subproblem, the entry corresponding to this box (i.e. \( a_{22} \)) has been disregarded. Therefore, 2 points have been awarded for this subproblem if all the other entries except \( a_{22} \) were correct.

(b) Consider the following Butcher tableau for a Runge-Kutta method:

\[
\begin{array}{c|ccc}
\frac{1}{2} + \frac{\sqrt{3}}{6} & \frac{1}{2} + \frac{\sqrt{3}}{6} & 0 \\
\frac{1}{2} - \frac{\sqrt{3}}{6} & -\frac{\sqrt{3}}{3} & \frac{1}{2} + \frac{\sqrt{3}}{6} \\
\end{array}
\]

Does this correspond to an explicit or an implicit method?

Solution: Implicit method. Points: [+1,0,-1]

This is because it does not have a strictly lower triangular matrix. In fact, this is Crouzeix’s two-stage, 3rd order Diagonally Implicit Runge Kutta method.

(c) Consider the following Butcher tableau for a Runge-Kutta method:

\[
\begin{array}{c|cccc}
\frac{1}{2} & 1 & 0 & 0 & 0 \\
\frac{3}{2} & \gamma & \frac{1}{2} & 0 & 0 \\
\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 0 \\
1 & \frac{3}{2} & -\frac{3}{2} & \frac{1}{2} & \frac{1}{2} \\
\end{array}
\]

\[
\begin{array}{c|cccc}
\frac{1}{2} & \delta & -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\
\end{array}
\]

What should be the values of \( \gamma \) and \( \delta \) such that the method is consistent?

**Solution:** (+1 for correct \( \gamma \), +1 for correct \( \delta \), no negatives)

\[
\gamma = \frac{1}{6} \quad \text{and} \quad \delta = \frac{3}{2}.
\]

For consistency, the following conditions need to be satisfied:

\[
\sum_{i=1}^{s} b_i = 1
\]

and

\[
c_i = \sum_{j=1}^{s} a_{ij}.
\]
6. *Stiffness and stability [4 points]*.

(a) Consider the second order, scalar ODE

\[ \ddot{y}(t) = -y(t) \]

and its equivalent system of first order ODEs

\[ \dot{z}(t) = A \, z(t), \]  

(1)

where \( t \geq 0, \, z(0) = (y(0), \dot{y}(0))^\top \) and \( A \in \mathbb{R}^{2 \times 2} \). Compute explicitly the entries of \( A \).

**Solution:** [1 if all correct, 0 otherwise]

\[ A = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \]

(b) Is the ODE (1) stiff (where \( A \) is the solution from Part (a))?  

**Solution:** no [1, 0, -1]

(c) Is the following ODE stiff for \( c \gg 1 \)?

\[ \dot{y}(t) = \begin{pmatrix} -c & 1 \\ -1 & -c \end{pmatrix} y(t) \]

**Solution:** yes [1, 0, -1]

(d) Which of the methods  

(i) explicit midpoint  
(ii) explicit Euler  
(iii) implicit Euler  

has the following stability region (shaded grey area):

![Stability Region Diagram](image)

**Solution:** (iii) [1, 0, -1]