**Übung 1** *Schurkomplement*
Sei $\mathbb{K} = \mathbb{R}$ oder $\mathbb{K} = \mathbb{C}$. Gegeben sei eine Blockpartitionierung einer Matrix $A \in \mathbb{K}^{n\times n}$ der Form

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix},$$

wobei die Teilmatrix $A_{11} \in \mathbb{K}^{p\times p}$ $(1 \leq p \leq n)$ invertierbar sei. Für die anderen Teilmatrizen gilt $A_{12} \in \mathbb{K}^{p\times(n-p)}$, $A_{21} \in \mathbb{K}^{(n-p)\times p}$ und $A_{22} \in \mathbb{K}^{(n-p)\times(n-p)}$. Die Block LR-Zerlegung von $A$ lässt sich dann darstellen als

$$A = \begin{pmatrix} Id & 0 \\ A_{21}A_{11}^{-1} & Id \end{pmatrix} \begin{pmatrix} A_{11} & A_{12} \\ 0 & S \end{pmatrix}$$

mit entsprechend dimensionierten Einheitsmatrizen $Id$. Die Teilmatrix $S \in \mathbb{K}^{(n-p)\times(n-p)}$ wird Schurkomplement von $A_{11}$ in $A$ genannt.

a) Zeigen Sie
$$S = A_{22} - A_{21}A_{11}^{-1}A_{12}.$$

b) Zeigen Sie: Wenn A hermitesch positiv definit ist, dann sind auch $A_{11}$ und $S$ hermitesch positiv definit.

**( 3 Punkte )**

**Übung 2** *LU-Zerlegung tridiagonaler Matrizen*
Gegeben sei eine Tridiagonalmatrix

$$A = \begin{pmatrix} a_1 & b_1 & & & 0 \\ c_2 & a_2 & b_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & b_{n-1} \\ 0 & & & c_n & a_n \end{pmatrix}$$

mit

$$\begin{aligned} &|a_1| > |b_1| > 0 \\ &|a_j| \geq |b_j| + |c_j| > 0, \quad b_j, c_j \neq 0, \quad j \in \{2, ..., n-1\} \\ &|a_n| \geq |c_n| > 0\ . \end{aligned}$$

Zeigen Sie: Der Algorithmus

$$\begin{aligned} &u_1 := a_1 \\ &l_j := c_j/u_{j-1} && j \in \{2, ..., n\} \\ &u_j := a_j - l_j b_{j-1} && j \in \{2, ..., n\} \end{aligned}$$

ist durchführbar (d.h. $u_1, \cdots, u_n \neq 0$) und liefert die LU-Zerlegung

$$A = \begin{pmatrix} 1 & & & 0 \\ l_2 & 1 & & \\ & \ddots & \ddots & \\ 0 & & l_n & 1 \end{pmatrix} \begin{pmatrix} u_1 & b_1 & & 0 \\ & u_2 & \ddots & \\ & & \ddots & b_{n-1} \\ 0 & & & u_n \end{pmatrix}$$

Zeigen Sie, dass $\det(A) \neq 0$ gilt (woraus die Invertierbarkeit von $A$ folgt).

**( 5 Punkte )**

**Übung 3** *LU-Zerlegung im Besonderen*
Sei $A \in \mathbb{R}^{n \times n}$ gegeben durch

$$a_{ij} = \begin{cases} +1 & \text{wenn } i = j \text{ oder } j = n \\ -1 & \text{wenn } i > j \\ 0 & \text{sonst.} \end{cases}$$

a) Begründen Sie, dass die LU-Zerlegung ohne Pivotisierung von $A$ die Eigenschaften

$$|l_{ij}| \leq 1 \qquad \text{und} \qquad u_{nn} = 2^{n-1}$$

erfüllt.

b) Zeigen Sie, dass eine LU-Zerlegung von $A$ mit Zeilen- und Spaltenvertauschung existiert mit

$$|u_{nn}| = 2 = \max_{i,j=1..n} \{u_{ij}\}.$$

Tipp: Betrachten Sie zunächst $n = 4$ um die beste Vertauschungsstrategie zu finden.

**( 5 Punkte )**

**Übung 4** *Lineare iterative Löser (Praktische Übung)*

In der Vorlesung haben Sie iterative Lösungsverfahren kennengelernt, die als Alternative zu direkten Lösern insbesondere bei dünnbesetzten Matrizen nennenswerte Vorteile versprechen. Lineare Iterationsverfahren haben die spezielle Form

$$x^{k+1} = x^k + W^{-1}(b - Ax^k),$$

und sie unterscheiden sich in der Wahl der Matrix $W$, die eine geeignete simple Approximation von $A$ darstellt. Hier werden Sie solche Löser selbst implementieren und zum Einsatz bringen. Die zu lösende Matrix $A$ finden Sie im vorgegebenen Programmgerüst.

- Implementieren Sie Richardson-, Jacobi- und Gauß-Seidel-Verfahren zur Lösung des Gleichungssystems. Beachten Sie:
  - Die Matrix $W$ könnte zwar explizit aufgesetzt und z.B. mit in `hdnum` bestehenden Lösern invertiert werden. Dabei kann es aber leicht passieren, dass das Invertieren von $W$ so rechenaufwendig ist wie es das direkte Invertieren von $A$ wäre. Entsprechend sollten Sie die jeweilige Struktur von $W$ nutzen um die Anwendung von $W^{-1}$ möglichst effizient zu realisieren.
  - Nutzen Sie als Abbruchkriterium für Ihr Verfahren eine Reduktion der Defektnorm (verglichen mit dem initialen Defekt $b - Ax^0$) um einen Faktor $10^{-4}$.
  - Je nach Matrix kann es sein, dass das Richardson-Verfahren praktisch nicht zum Konvergieren gebracht werden kann.
  - Testen Sie auf geeignete Weise, ob Ihre Implementierung tatsächlich die Lösung approximiert. Dieser Test soll Teil der Abgabe sein.
- Vergleichen Sie die benötigte Zeit Ihrer Löser mit einem direkten LU-Löser, den Sie über die Funktion `linsolve` nutzen können (Vorsicht: Diese Funktion überschreibt ihre Argumente!). Plotten Sie dann die benötigte Zeit der verschiedenen Verfahren gegen Matrixgrößen $N \times N$, wobei $N = 2^n$ mit $n \in \{4, ..., 9\}$.

- Um die Skalierbarkeit Ihrer Löser zu verbessern, nutzen Sie nun für Matrix-Vektor-Produkte $Ax$ zusätzlich eine Kopie der Matrix $A$ in Form der vorgegebenen `SparseMatrix`. Hier können Sie entweder Ihre eigene Implementierung der vorangegangenen Aufgabe nutzen oder die im Programmgerüst Vorgegebene. Diese einfache Implementierung einer dünnbesetzten Matrix erlaubt keinen direkten Zugriff auf Matrixelemente, daher können Sie für die Umsetzung von $W^{-1}$ weiterhin die bisherige `DenseMatrix` $A$ nutzen. Weiterhin können Sie in Ihrer Implementierung von Gauß-Seidel ausnutzen, dass Einträge von $A$ weiter abseits der Diagonalen Null sind, um auch hier lineare Skalierbarkeit zu erreichen. Plotten Sie diese Resultate gegen die vorigen Durchläufe.

**( 3+1+1 Punkte )**