(32) Schreiben Sie eine Matlab–Funktion `x = rsolve(R, b)`, die ein lineares Gleichungssystem $Rx = b$ lst, wobei R eine obere Dreiecksmatrix sei.

(33) Schreiben Sie eine Matlab–Funktion `x = lsolve(L, b)`, die ein lineares Gleichungssystem $Lx = b$ lst, wobei L eine untere Dreiecksmatrix sei.

(34) Testes Sie die Funktionen aus den Beispielen 32 und 33, indem Sie jeweils zehn dreieckige Zufallsmatrizen der Dimensionen 1 bis 100 erzeugen und das Ergebnis mit dem Resultat vergleichen, das Sie mit dem \–Operator von Matlab berechnen. Überprüfen Sie die Aussage über den Aufwand $O(n^2)$ aus der Vorlesung empirisch.

(35) Berechnen Sie mit der Hand eine LR–Zerlegung für die Matrix

$$A = \begin{pmatrix} 1 & 4 & 3 \\ 2 & 6 & 5 \\ 3 & 1 & -4 \end{pmatrix}.$$

(36) Verwenden Sie die Zerlegung aus Beispiel 35, um das lineare Gleichungssystem $Ax = b$ mit dem Vektor

$$b = \begin{pmatrix} 2 \\ 1 \\ -4 \end{pmatrix}$$

zu lösen.

(37) Schreiben Sie eine Funktion `[P, L, R] = lr(A)`, die eine LR–Zerlegung der Matrix A berechnet.

(38) Testen Sie die Funktion aus Beispiel 37 an jeweils zehn Zufallsmatrizen der Dimensionen 1 bis 100, und überprüfen Sie wieder den asymptotischen Aufwand $O(n^3)$.

(39) Schreiben Sie eine Matlab–Funktion `x = lrsolve(A, b)`, die mit Hilfe der Funktionen `lr`, `lsolve` und `rsolve` ein lineares Gleichungssystem $Ax = b$ löst.

(40) Erzeugen Sie wieder jeweils zehn Zufallsmatrizen und Zufallsvektoren der Dimensionen 1 bis 100, testen Sie die Funktion aus Beispiel 39, indem Sie mit dem \–Operator von Matlab vergleichen, und untersuchen Sie wieder den asymptotischen Aufwand.