Fachbereich Mathematik Sommersemester 2022
Thomas Markwig

Algorithmen der Linearen Algebra

1 Erste Schritte in SINGULAR

Aufgabe 1:
Starte SINGULAR , berechne die Summe der ganzen Zahlen 4 und 5 und beende SINGULAR wieder.

Aufgabe 2:
Definiere in SINGULAR zwei Variablen `x` und `y` vom Typ `int` und berechne ihr Produkt.

Aufgabe 3:
Berechne in SINGULAR die folgende Summe von ganzzahligen Matrizen:

$$\begin{pmatrix} 1 & 2 & 5 \\ -1 & 0 & 3 \end{pmatrix} + \begin{pmatrix} -1 & 0 & 2 \\ 1 & 1 & 1 \end{pmatrix}.$$

Aufgabe 4:
Berechne in SINGULAR die folgende Produkt von ganzzahligen Matrizen:

$$\begin{pmatrix} 1 & 2 & 5 \\ -1 & 0 & 3 \end{pmatrix} \circ \begin{pmatrix} -1 & 0 \\ 2 & 1 \\ 1 & 1 \end{pmatrix}.$$

Aufgabe 5:
Definiere in SINGULAR eine Variable `R` vom Typ `ring` und weise ihr als Wert den Ring $\mathbb{Q}[t]$ zu. Lasse den Wert von `R` anzeigen.

Aufgabe 6:
Definiere in SINGULAR eine Variable `f` vom Typ `poly` im Ring `R` und weise ihr den Wert $t^4 + 2t - 1$ zu.

Aufgabe 7:
Werte in SINGULAR das Polynom `f` aus der vorigen Aufgabe an der Stelle $t = 2$ aus. Verwende dazu den Befehl `subst` und schaue zunächst dessen Syntax in der Hilfeanweisung von SINGULAR nach.

Aufgabe 8:
Definiere in SINGULAR eine Variable `M` vom Typ `matrix` und weise ihr folgende Matrix als Wert zu:

$$\begin{pmatrix} 4 & t^2 & f \\ 0 & 1 & 0 \end{pmatrix}$$

Addiere dann die Matrix M zu einer zuvor definierten Matrix vom Typ `intmat`. Versuche, das Ergebnis einer Variablen vom Typ `intmat` und einer vom Typ `matrix` zuzuordnen. Wie ist das Ergebnis zu verstehen?

Aufgabe 9:
Ziehe in SINGULAR aus der Matrix `M` die zweite Zeile heraus und ordne sie einer neuen Variablen vom Typ `matrix` der Größe 1×3 zu.

Aufgabe 10:
Berechne in SINGULAR das Produkt der natürlichen Zahlen von 1 bis 10 mittels mit Hilfe einer `for`-Schleife.

Aufgabe 11:
Multipliziere in SINGULAR die natürlichen Zahlen von 1 an, bis da Ergebnis den Wert 500 überschreitet. Bei welcher natürlichen Zahl ist dies der Fall und wie groß ist das Produktdann. Hierfür eignet sich eine `while`-Schleife besonders gut.

2 Erste Algorithmen und Prozeduren

Konvention 1
Wenn bei Prozeduren eine Reihenfolge der Parameter angegeben ist, die eingelesen werden sollen, dann ist diese bei der Implementierung in SINGULAR einzuhalten. Wenn Unklarheiten bestehen, sollte nachgefragt werden.

Aufgabe 12:
Schreibe eine Prozedur `quadratsumme` unter Verwendung einer Schleife, die eine natürliche Zahl `n` einliest und die Summe der Quadratzahlen $1^2, 2^2, 3^2, \ldots, n^2$ ausgibt. Formuliere zunächst den Algorithmus hierfür.

Aufgabe 13:
Schreibe eine neue, rekursive Prozedur `quadratsumme_rekursiv`, die die Summe der ersten n Quadratzahlen berechnet. Formuliere zunächst den Algorithmus hierfür.

Aufgabe 14:
Schreibe eine Prozedur `binomi`, die zwei natürliche Zahlen `n` und `k` einliest und den Binomialkoeffizienten $\binom{n}{k}$ zurückgibt. Formuliere zunächst den Algorithmus hierfür. (Vereinbarung: falls $k < 0$ oder $k > n$, dann $\binom{n}{k} = 0$.)

Aufgabe 15:
Schreibe eine Prozedur `minimum`, die einen Vektor von natürlichen Zahlen einliest und das Minimum der Zahlen ausgibt. Formuliere zunächst den Algorithmus hierfür.

Aufgabe 16:
Schreibe `Example`-Teile für die Prozeduren in den letzten drei Aufgaben.

Aufgabe* 17: Schreibe eine Prozedur `abs_val`, die den Absolutbetrag einer reellen Zahl ausgibt.

Aufgabe* 18: Schreibe Prozeduren `zeilensummennorm`, `maximumsnorm` und `q_eukl_norm`, die eine $(m \times n)$-Matrix `A` von reellen Zahlen einlesen und

(a) die Zeilensummennorm von `A` (d. h. $\max_{i=1,\dots,m}\left(\sum_{j=1}^{n}|A_{ij}|\right)$),

(b) die Maximumsnorm von `A` (d. h. $\max\left(|A_{ij}| \mid i = 1,\dots,m,\ j = 1,\dots,n\right)$), respektive

(c) das Quadrat der euklidischen Norm berechnen (d. h. $\sum_{i,j}|A_{ij}|^2$).

Für den Absolutbetrag verwende die Funktion `abs` aus der Bibliothek `linalg.lib`. Formuliere jeweils zunächst den Algorithmus.

3 Der Gauß-Algorithmus

Aufgabe 19:
Schreibe eine Prozedur `zeilentausch`, die eine Matrix der Größe $m \times n$-einliest sowie zwei ganze Zahlen $1 \leq i, j \leq m$, die Zeilen i und j tauscht und das Ergebnis zurück gibt. Formuliere zunächst den Algorithmus hierfür.

Aufgabe 20:
Schreibe eine Prozedur `zeilenvielfaches`, die eine Matrix der Größe $m \times n$-einliest sowie eine ganze Zahl $1 \leq i \leq m$ und ein Polynom f, die Zeile i mit f multipliziert und das Ergebnis zurück gibt. Formuliere zunächst den Algorithmus hierfür.

Aufgabe 21:
Schreibe eine Prozedur `elementarezeilenoperation`, die eine Matrix der Größe $m \times n$-einliest sowie zwei Zahlen $1 \leq i, j \leq n$ und ein Polynom f, die dann das f-fache der j-ten Zeile zur i-ten Zeile addiert und das Ergebnis zurück gibt. Formuliere zunächst den Algorithmus hierfür.

Aufgabe 22:
Schreibe Prozeduren `spaltentausch`, `spaltenvielfaches` und `elementarespaltenoperation`, die die entsprechenden Spaltenoperationen an einer Matrix durchführen. Dabei darf der SINGULAR-Befehl `transpose` verwendet werden.

Aufgabe 23:
Schreibe eine rekursive Prozedur `gauss`, die eine Matrix A einliest und die mittels Gauß-Elimination ermittelte Zeilen-Stufen-Form der Matrix ausgibt. Die Einträge der Matrizen sollen vom Typ `poly` sein. Teste Deine Ergebnisse mit der Prozedur `gauss_nf` aus der Bibliothek `linalg.lib`.

Aufgabe 24:
Schreibe eine Prozedur `rZSF`, die eine Matrix A einliest und die reduzierte Zeilen-Stufen-Form der Matrix ausgibt. Die Einträge der Matrizen sollen vom Typ `poly` sein.

4 Erste Anwendungen des Gauß-Algorithmus'

Aufgabe 25:
Schreibe eine Prozedur `rang`, die eine Matrix A einliest und ihren Rang ausgibt.

Aufgabe 26:
Schreibe eine Prozedur `invers`, die eine quadratische Matrix einliest und ihre Inverse ausgibt, falls sie invertierbar ist.

Aufgabe* 27: Schreibe eine Prozedur `matrixNF`, die für eine Matrix A ihre Normalform sowie zwei invertierbare Matrizen S und T berechnet, so daß $S \circ A \circ T$ die Normalform ist.

Konvention 2
Vektoren im K^n werden als Matrizen der Größe $n \times 1$ eingegeben. Familien von k Vektoren im K^n werden als Matrizen der Größe $n \times k$ an die Prozeduren übergeben, bei denen die Spalten die Vektoren in der Familie sind.

Aufgabe 28:
Schreibe eine Prozedur `basis`, die für eine Familie von Vektoren in K^n eine Basis für den von diesen erzeugten Unterraum berechnet. Die Familie von Vektoren soll als Spalten einer Matrix übergeben werden.

Aufgabe* 29: Schreibe eine Prozedur `injektiv_surjektiv`, die eine Matrix A einliest und überprüft, ob f_A injektiv und / oder surjektiv ist.

Aufgabe* 30: Schreibe eine Prozedur `summe`, die zwei Familien von Vektoren im K^n einliest und eine Basis der Summe der von ihnen erzeugten Unterräume ausgibt.

Aufgabe* 31: Schreibe eine Prozedur `linear_unabhaengig`, die eine Familie von Vektoren im K^n auf lineare Unabhängigkeit überprüft. Die Ausgabe sollte 0 oder 1 sein, je nachdem ob die Familie lineare unabhängig ist oder nicht.

Aufgabe* 32: Schreibe eine Prozedur `bild`, die eine Matrix A einliest und eine Basis des Bildes von f_A ausgibt.

5 Lineare Gleichungssysteme und Anwendungen

Aufgabe 33:
Schreibe eine Prozedur `lgs`, die eine erweiterte Koeffizientenmatrix einliest und eine spezielle Lösung sowie eine Basis des Lösungsraums des homogenen LGS berechnet und ausgibt, sofern das LGS lösbar ist. Wenn das LGS lösbar ist, soll die Ausgabe eine Liste sein, die in der ersten Komponente die spezielle Lösung

enthält und in der zweite Komponente die Basis des Lösungsraums des homogenen LGS.

Aufgabe 34:
Schreibe eine Prozedur `kern`, die eine Matrix A einliest und eine Basis des Kerns von f_A ausgibt.

Aufgabe* 35: Schreibe eine Prozedur `basistransformation`, die zwei Basen B und B' des K^n einliest und die Transformationsmatrix $T^B_{B'}$ ausgibt.

Aufgabe 36:
Schreibe eine Prozedur `matrixdarstellung`, die zwei Basen B des K^n und D des K^m sowie eine Matrix $A \in \text{Mat}(m \times n, K)$ einliest und die Matrixdarstellung $M^B_D(f_A)$ ausgibt.

Aufgabe* 37: Schreibe eine Prozedur `steinitz`, die eine Basis von $V \leq K^n$ und eine linear unabhängige Famile F in V einliest und eine Basis V ausgibt, die sich aus F und weiteren Vektoren von B zusammensetzt.

Aufgabe* 38: Schreibe eine Prozedur `gleichungen`, die eine Familie F von Vektoren im K^n einliest und eine Matrix $A \in \text{Mat}(m \times n, K)$ berechnet, so daß $\text{Lös}(A, 0) = \text{Lin}(F)$ gilt.

Aufgabe* 39: Schreibe eine Prozedur `durchschnitt`, die zwei Familien von Vektoren im K^n einliest und eine Basis des Durchschnitts der von diesen erzeugten Unterräume berechnet.

6 Die Determinante

Aufgabe 40:
Schreibe eine Prozedur `determinante`, die die Determinante einer Matrix $A \in \text{Mat}_n(K)$ mit Hilfe des Gauß-Algorithmus' berechnet. Formuliere den Algorithmus dazu in rekursiver Form.

Aufgabe 41:
Schreibe eine Prozedur `determinante_laplace`, die die Determinante einer Matrix $A \in \text{Mat}_n(R)$ mit Hilfe des Laplaceschen Entwicklungssatzes mit Einwicklung nach der ersten Spalte berechnet.

Aufgabe* 42: Formuliere einen Algorithmus, der ein Gleichungssystem mit Hilfe der Cramerschen Regel löst. Schreibe die zugehörige Prozedur `cramer`, die die erweiterte Koeffizientenmatrix des LGS als Eingabe erhält und die eindeutige Lösung zurück gibt.

7 Endomorphismen und die Jordansche Normalform

Aufgabe 43:
Schreibe eine Prozedur `charakteristisches_polynom`, die das charakteristische Polynom einer Matrix $A \in \mathrm{Mat}_n(K)$ berechnet. Als Variable soll die erste Variable des Rings genommen werden, auf die mittels der Funktion `var` zugegriffen werden kann.

Aufgabe 44:
Schreibe eine Prozedur `minimal_polynom`, die das Minimalpolynom einer Matrix $A \in \mathrm{Mat}_n(K)$ berechnet. Als Variable soll die erste Variable des Rings genommen werden, auf die mittels der Funktion `var` zugegriffen werden kann.

Aufgabe 45:
Schreibe eine Prozedur `eigenraum`, die eine Matrix A und einen Eigenwert λ von A einliest und eine Basis des Eigenraums $\mathrm{Eig}(A, \lambda)$ ausgibt.

Aufgabe 46:
Schreibe eine Prozedur `diagonalisierbar`, die für eine quadratische Matrix $A \in \mathrm{Mat}_n(\mathbb{Q})$ feststellt, ob sie über $\mathbb{C}$ diagonalisierbar ist.

Aufgabe 47:
Schreibe eine Prozedur `diagonalisierung`, die für eine quadratische Matrix $A \in \mathrm{Mat}_n(K)$ feststellt, ob sie über K diagonalisierbar ist, und ggf. eine invertierbare Matrix T berechnet, so daß $T^{-1} \circ A \circ T$ eine Diagonalmatrix ist. Rückgabe sollte eine Liste sein, deren erster Eintrag 1 oder 0 ist, je nachdem ob die Matrix diagonalisierbar ist oder nicht, und deren zweiter Eintrag im Falle der Diagonalisierbarkeit die Matrix T enthält.

Aufgabe* 48: Schreibe eine Prozedur `eigenwerte`, die die Eigenwerte einer Matrix $A \in \mathrm{Mat}_n(\mathbb{C})$ näherungsweise bestimmt.

Aufgabe 49:
Formulieren einen Algorithmus und schreibe eine Prozedur `elementarteiler`, die die Eigenwerte und die zugehörigen Elementarteiler einer Matrix berechnet, wenn deren charakteristisches Polynom zerfällt.

Aufgabe 50:
Schreibe eine Prozedur `jnf`, die die Jordansche Normalform und die zugehörige Transformationsmatrix einer Matrix $A \in \mathrm{Mat}_n(K)$ berechnet, wenn deren charakteristisches Polynom zerfällt, und diese als Liste zurückgibt.

Aufgabe 51:
Sei $p \in K[t]$ ein festes Polynom vom Grad $d \geq 1$, dann ist der Einsetzhomomorphismus

$$\Phi_p : K[t]_{\leq n} \longrightarrow K[t]_{\leq d \cdot n} : f \mapsto f(p)$$

eine K-lineare Abbildung des Vektorraums der Polyome vom Grad höchstens n in den Vektorraum der Polynome vom Grad höchstens $d \cdot n$. Wir betrachten die Vektorräume mit den kanonischen Basen $B = (1, t, \ldots, t^n)$ und $D = (1, t, \ldots, t^{d \cdot n})$. Schreibe eine Prozedur `MD_einsetzhom`, die ein positive Zahl n und ein Polynom $p \in K[t] \setminus K$ einliest und die Matrixdarstellung $M_D^B(\Phi_p)$ ausgibt. Man greife dabei mittels `var(1)` auf die erste Variable des Ringes zurück und erwarte nicht, dass diese `t` heißt.

Aufgabe 52:
Schreibe eine Prozedur `spur_einsetzhom`, die eine natürliche Zahl n und ein lineares Polynom $p = a \cdot t + b \in K[t]$ einliest und die Spur des Einsetzhomomorphismus Φ_p berechnet.

8 Der Euklidische Algorithmus und seine Anwendungen

Aufgabe 53:
Schreibe eine Prozedur `ggt`, die zwei ganze Zahlen oder zwei Polynome ungleich Null einliest und den positiven größten gemeinsamen Teiler bzw. den normierten gemeinsamen Teiler der beiden ausgibt. Man verwende die Befehle `div` und `mod`. Ferner nutze man den Datentyp `def` für die Eingabe und die Funktion `typeof`, um festzustellen, ob es sich bei der Eingab um Polynome oder ganze Zahlen handelt.

Aufgabe 54:
Schreibe eine rekursive Prozedur `extggt`, die zwei ganze Zahlen oder zwei Polynome a und b ungleich Null einliest und den positiven größten gemeinsamen Teiler g bzw. den normierten gemeinsamen Teiler g der beiden sowie zwei ganze Zahlen bzw. Polynome x und y mit

$$g = x \cdot a + y \cdot b$$

ausgibt.

Aufgabe 55:
Beschreibe einen Algorithmus, der überprüft, ob ein Element $\overline{a} \in \mathbb{Z}_n$ invertierbar ist und ggf. das Inverse ausgibt (siehe [Mar08, Prop. 7.56]. Schreibe eine Prozedur `inv_mod_n`, die n und a einliest und das Inverse von $\overline{a} \in \mathbb{Z}_n$ ausgibt, wenn es existiert.

Aufgabe 56:
Schreibe eine Prozedur `chinesischer_restsatz`, die zwei Vektoren $a = (a_1, \ldots, a_r)$ und $n = (n_1, \ldots, n_r)$ ganzer Zahlen einliest, wobei die n_i paarweise teilerfremd sind, und dann eine ganze Zahl $0 \leq x < n_1 \cdot \ldots \cdot n_r$ ausgibt, für die

$$x \equiv a_i \pmod{n_i}$$

für alle $i = 1, \ldots, r$ gilt.

9 Euklidische Räume

Aufgabe 57:
Schreibe eine Prozedur `sqrt_heron`, die eine positive rationale Zahl c und eine positive ganze Zahl n einliest und eine rationale Zahl q ausgibt, so daß c und q^2 bis auf n Dezimalstellen übereinstimmen.

Aufgabe 58:
Schreibe eine Prozedur `positiv_definit`, der eine Matrix in $\mathrm{Mat}_n(\mathbb{R})$ einliest und überprüft, ob diese symmetrisch und positiv definit ist. Falls ja, soll 1 zurückgegeben werden, sonst 0.

Aufgabe 59:
Schreibe eine Prozedur `OGB`, die eine Familie von Vektoren im $\mathbb{R}^n$ sowie eine positiv definite symmetrische Matrix $A \in \mathrm{Mat}_n(\mathbb{R})$ einliest und eine Orthogonalbasis von $\mathrm{Lin}(M)$ bezüglich des Skalarproduktes b_A zurückgibt.

Aufgabe 60:
Schreibe eine Prozedur `ONB`, die eine Familie von Vektoren im $\mathbb{R}^n$ sowie eine positiv definite symmetrische Matrix $A \in \mathrm{Mat}_n(\mathbb{R})$ einliest und eine Orthonormalbasis von $\mathrm{Lin}(M)$ bezüglich des Skalarproduktes b_A zurückgibt. Dabei sollen die Einträge der Vektoren mit einer Präzision von 15 Nachkommastellen berechnet worden sein.

Aufgabe 61:
Beschreibe einen Algorithmus, der überprüft, ob eine Matrix $A \in \mathrm{Mat}_n(\mathbb{C})$ normal ist. Schreibe einen Algorithmus `normal`, der den Algorithmus implementiert und 1 zurück gibt, wenn die Matrix normal ist, und 0 sonst.

Aufgabe 62:
Schreibe einen Prozedur `normal_diagonal`, die für eine normale Matrix $A \in \mathrm{Mat}_n(\mathbb{Q})$ deren charakteristisches Polynom über $\mathbb{Q}$ zerfällt, eine Diagonalmatrix D sowie eine orthogonale Matrix T bestimmt mit

$$D = T^{-1} \circ A \circ T.$$

Dabei sollen die Einträge in T mit einer Präzision von 15 Nachkommastellen berechnet worden sein.

Aufgabe 63:
Schreibe eine Prozedur `eigenwerte_symmetrisch`, die für eine symmetrische Matrix $A \in \mathrm{Mat}_n(\mathbb{Q})$ die Eigenwerte approximiert und ihre Vielfachheiten exakt bestimmt. Beschreibe zunächst den zugehörigen Algorithmus.

10 Multilineare Algebra

Aufgabe 64:
Schreibe eine Prozedur `symmetrischer_gauss`, die eine symmetrische Matrix $A \in \mathrm{Mat}_n(K)$ einliest und, wenn $\mathrm{char}(K) \neq 2$, eine invertierbare Matrix $T \in \mathrm{Gl}_n(K)$ berechnet, so daß $T^t \circ A \circ T$ eine Diagonalmatrix ist.

Aufgabe 65:
Schreibe eine Prozedur `sylvester`, die für eine symmetrische Matrix $A \in \mathrm{Mat}_n(\mathbb{R})$ den Trägheitsindex, den Morseindex und die Signatur berechnet. Beschreibe zunächst den zugehörigen Algorithmus (siehe [Mar20, Kor. 18.28]).

Aufgabe 66:
Schreibe eine Prozedur `duale_basis`, die eine Basis von K^n einliest und die duale Basis ausgibt.

Aufgabe 67:
Schreibe eine Prozedur `reiner_tensor`, die eine Matrix $A \in \mathrm{Mat}(m \times n, K)$ vom Rang 1 einliest und zwei Vektoren $u \in K^m$ und $v \in K^n$ ausgibt, so daß $A = u \circ v^t$.

Aufgabe 68:
Schreibe eine Prozedur `minimale_tensorzerlegung`, die eine Matrix $A \in \mathrm{Mat}(m \times n, K)$ einliest und $r = \mathrm{rang}(A)$ Matrizen $A_1, \ldots, A_r$ vom Rang 1 ausgibt, so daß $A = A_1 + \ldots + A_r$.

Aufgabe* 69: Schreibe eine Prozedur `tensorprodukt_matrizen`, die zwei Matrizen $A \in \mathrm{Mat}(n' \times n, K)$ und $B \in \mathrm{Mat}(m' \times m, K)$ einliest und das Tensorprodukt $A \otimes B$ und damit $\mathrm{Im}(f_A) \otimes \mathrm{Im}(f_B)$ berechnet. Formuliere zunächst den Algorithmus (siehe [Mar20, Bsp. 20.22+20.24]).

Aufgabe* 70: Schreibe eine Prozedur `minor`, die eine Matrix $A \in \mathrm{Mat}(m \times n, K)$ sowie zwei ganzzahlige Vektoren $(i_1, \ldots, i_r)$ und $(j_1, \ldots, j_r)$ mit $1 \leq i_1 < \ldots < i_r \leq m$ und $1 \leq j_1 < \ldots < j_r \leq n$ einliest und den Minor $A(i_1, \ldots, i_r | j_1, \ldots, j_r)$ berechnet, d.h. die Determinante der Matrix, die aus A entsteht, wenn ausser den Zeilen $i_1, \ldots, i_r$ und den Spalten $j_1, \ldots, j_r$ alle Zeilen und Spalten gestrichen werden. Formuliere zunächst eine Algorithmus hierfür.

Aufgabe* 71: Schreibe eine Prozedur `MD_auesseres_produkt`, die eine Matrix $A \in \mathrm{Mat}(m \times n, K)$ sowie eine ganze Zahl $1 \leq r \leq \min\{m, n\}$ einliest und die Matrixdarstellung von $\bigwedge^r f_A$ bezüglich der kanonischen Basen auf den äußeren Produkten ausgibt. Formuliere zunächst den Algorithmus (siehe [Mar20, Prop. 21.16]).

Aufgabe* 72: Leite aus dem Allgemeinen Laplaceschen Entwicklungssatz (siehe [Mar20, Satz 21.34]) einen Algorithmus zur Berechnung der Determinante her und schreibe eine Prozedur, die eine quadratische Matrix $A \in \mathrm{Mat}_n(K)$ sowie eine Partition $\{1, \ldots, n\} = \{i_1, \ldots, i_r\} \cup \{j_1, \ldots, j_{n-r}\}$ einliest und die Determinante von A berechnet.

Literatur

[Mar08] Thomas Markwig, *Algebraische Strukturen*, Vorlesungsskript, TU Kaiserslautern, 2008.

[Mar20] ______, *Lineare Algebra*, Vorlesungsskript, Universität Tübingen, 2020.