4. Seznamy a lineární algebra

Seznamy se zapisují ve tvaru {x_1,x_2,...,x_n}.
První a poslední prvek seznamu: `First[seznam]` `Last[seznam]`
Přístup k prvku seznamu na pozici i: `seznam[[i]]`
Podseznam vymezený pozicemi i a j: `seznam[[i;;j]]`
Sestavení seznamu z funkčních hodnot: `Table[f[i],{i,min,max}]` `Table[f[i],{i,min,max,krok}]`
Seznam s identickými prvky: `Table[prvek,{počet kopií}]`
Aritmetické posloupnosti lze generovat pomocí `Range[n]`, `Range[min,max]`, `Range[min,max,krok]`.

Vložení prvku na konec seznamu: `Append[seznam,prvek]` `AppendTo[seznam,prvek]`
Vložení prvku na začátek seznamu: `Prepend[seznam,prvek]` `PrependTo[seznam,prvek]`
Vložení prvku na zadanou pozici v seznamu: `Insert[seznam,prvek,pozice]`
Smazání prvku seznamu na zadané pozici: `Delete[seznam,pozice]`
Délka seznamu: `Length[seznam]`
Součet prvků seznamu: `Total[seznam]`
Obrácení seznamu: `Reverse[seznam]`
Setřídění seznamu: `Sort[seznam]`
Cyklický posun seznamu: `RotateLeft[seznam]` `RotateLeft[seznam,n]` (analogicky `RotateRight`)
Spojení seznamů: Join[$\texttt{seznam}_1$,$\texttt{seznam}_2$,...]
Sjednocení (spojení, odstranění duplicit a setřídění): Union[$\texttt{seznam}_1$,$\texttt{seznam}_2$,...]
Průnik: Intersection[$\texttt{seznam}_1$,$\texttt{seznam}_2$,...]
Rozdíl $S\backslash(S_1 \cup S_2 \cup \cdots)$: Complement[S,$S_1$,$S_2$,...]

Každou funkci f, která má nastaven atribut `Listable` (viz `Attributes[f]`), lze aplikovat na seznamy:
Funkce jedné proměnné: f[{x_1,x_2,...}] $\to$ {f[x_1],f[x_2],...}
Funkce více proměnných: f[{x_1,x_2,...},{y_1,y_2,...},...] $\to$ {f[x_1,y_1,...],f[x_2,y_2,...],...}

Vektory z $\mathbf{R}^n$ nebo $\mathbf{C}^n$ se zapisují jako n-prvkové seznamy {x_1,x_2,...,x_n}.
Násobení vektoru skalárem: a{x_1,x_2,...}
Skalární součin: {x_1,x_2,...}.{y_1,y_2,...}
Vektorový součin v $\mathbf{R}^3$: Cross[{x_1,x_2,x_3},{y_1,y_2,y_3}]
Eukleidovská norma: `Norm[vektor]`

Matice $\{a_{ij}\}$ typu $m \times n$ je v Mathematice reprezentována jako seznam řádkových vektorů, tj. ve tvaru $\{\{a_{11}, \dots, a_{1n}\}, \dots, \{a_{m1}, \dots, a_{mn}\}\}$. Tento seznam lze zobrazit jako matici pomocí `MatrixForm`.
Přístup k prvku matice na pozici (i, j): `A[[i,j]]`
Sestavení matice z funkčních hodnot: Table[f[i,j],{i,$\texttt{min}_1$,$\texttt{max}_1$},{j,$\texttt{min}_2$,$\texttt{max}_2$}]
Jednotková matice řádu n: `IdentityMatrix[n]`
Diagonální matice: `DiagonalMatrix[seznam prvků na diagonále]`
Transponovaná matice: `Transpose[matice]`
Násobení matic: `A.B`
Inverzní matice: `Inverse[matice]`
Determinant: `Det[matice]`
Mocnina matice: `MatrixPower[matice,exponent]`
Hodnost matice: `MatrixRank[matice]`
Gaussova eliminace: `RowReduce[matice]`
Řešení soustavy $Ax = b$: `LinearSolve[A,b]`
Báze všech řešení homogenní soustavy $Ax = 0$: `NullSpace[A]`
Vlastní čísla a vlastní vektory: `Eigenvalues[A]` `Eigenvectors[A]`
U funkcí `Inverse`, `Det`, `MatrixRank`, `RowReduce`, `LinearSolve` a `NullSpace` lze použít volbu `Modulus->p`.

Cvičení

1. Zkonstruujte seznam všech čísel z množiny $\{1, \ldots, 100\}$, které zbydou po vyškrtání všech násobků dvojky, trojky a pětky. Kolik takových čísel je?

2. Funkce `Divisors[n]` vrací seznam dělitelů čísla n. Najděte všechny společné dělitele čísel 2010 a 4140.

3. Zjistěte, zda jsou vektory $(0, -4, 3, 3)$, $(8, 1, 3, -3)$, $(1, 0, 0, 7)$ a $(1, 3, 4, 8)$ lineárně závislé.

4. Najděte inverzní matici k $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$. Zkontrolujte, že AA^{-1} je jednotková matice.

Návod: Zjednodušte součin matic pomocí `Simplify`.

5. Nadefinujte funkci `pridejSloupec[A,b]`, která pro zadanou matici $A \in \mathbf{R}^{m\times n}$ a vektor $b \in \mathbf{R}^m$ vyrobí matici typu $m \times (n + 1)$; ta vznikne připojením vektoru b jako posledního sloupce k matici A.

6. Uvažujme soustavu rovnic

$$\begin{aligned} x + 2y + 3z + w &= 3, \\ x + 4y + 5z + 2w &= 2, \\ 2x + 9y + 8z + 3w &= 7, \\ 5x + 7y + 9z + 2w &= 20. \end{aligned}$$

Nechť A je matice této soustavy a b je vektor pravé strany. Soustava má nekonečně mnoho řešení (spočítejte hodnost A), funkce `LinearSolve[A,b]` vrátí pouze jedno (partikulární) řešení; proveďte zkoušku (vynásobte matici A řešením). Všechna řešení lze získat pomocí `Reduce[A.{x,y,z,w}==b,{x,y,z,w}]`, vyzkoušejte. Pomocí `NullSpace[A]` najděte bázi řešení homogenní soustavy a podívejte se, jak výsledek souvisí s dříve nalezeným obecným řešením nehomogenní soustavy.

7. Hilbertova matice řádu n je definována předpisem $H_n = \{\frac{1}{i+j-1}\}_{i,j=1}^{n}$. Nadefinujte funkci `hilbert[n]`, která pro zadané $n \in \mathbf{N}$ zkonstruuje matici H_n. Jak dlouho trvá výpočet H_{100}^{-1}? Použijte příkaz `AbsoluteTiming[Inverse[hilbert[100]];]` (středník zamezí zobrazení spočtené matice).

8. Otočení v rovině kolem bodu $[0, 0]$ o úhel α je lineární zobrazení s maticí $\begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix}$. Uvažujme elipsu s parametrizací $x(t) = 2\cos t$, $y(t) = \sin t$, $t \in [0, 2\pi]$. Vykreslete tuto elipsu do jednoho obrázku společně s elipsami, které vzniknou otočením o úhly $k\pi/4$, $k \in \{1, \ldots, 3\}$.

9. Latinský čtverec řádu n je tabulka tvořená $n \times n$ políčky. V každém z nich je zapsán jeden z n různých symbolů tak, že v žádném řádku ani sloupci se symboly neopakují. Položíme-li přes sebe dva latinské čtverce stejného řádu, dostaneme n^2 dvojic symbolů. Jsou-li všechny tyto dvojice navzájem různé, pak říkáme, že dané latinské čtverce jsou ortogonální.
Nechť p je prvočíslo. Pak pro každou volbu $a \in \{1, \ldots, p-1\}$ představuje matice $\{(a \cdot i + j) \mod p\}_{i,j=0}^{p-1}$ latinský čtverec řádu p a čtverce odpovídající různým volbám a jsou navzájem ortogonální. Nadefinujte funkci, která pro zadané p tímto způsobem zkonstruuje $p-1$ latinských čtverců řádu p, z nichž každé dva jsou ortogonální. Použijte ji k nalezení šesti navzájem ortogonálních latinských čtverců řádu 7. Vypište výsledek v přehledné podobě pomocí `MatrixForm`.

10. Nadefinujte funkci, která ze zadaného seznamu $\{a_1, \ldots, a_n\}$ vyrobí matici

$$\begin{pmatrix} a_1 & a_2 & \cdots & a_{n-1} & a_n \\ a_2 & a_3 & \cdots & a_n & a_1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_n & a_1 & \cdots & a_{n-2} & a_{n-1} \end{pmatrix}.$$

Použijte ji na seznam $\{1, \ldots, 10\}$.

11. Zkuste počítat mocniny matice $\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$. Uhodnete, jaký tvar má n-tá mocnina?

12. *Mathematica* umí provádět Gramovu-Schmidtovu ortogonalizaci, převádět matici na Jordanův kanonický tvar, počítat maticovou exponenciálu, hledat rozklady matic, atd. Vyhledejte některé z těchto operací v nápovědě (sekce *Matrices & Linear Algebra*) a vyzkoušejte, jak fungují.