
WPROWADZENIE DO ŚRODOWISKA R

WYMAGANIA: ŚRODOWISKO R, BIBLIOTEKI: dplyr, ggplot2, moments, stats (<https://cran.r-project.org>)

ZAD. 1. Oblicz wartości następujących wyrażeń w środowisku R:

a) $2 \cdot 3^3 + \log_7 2401$; b) $\sqrt[3]{2}$; c) $\sqrt[5]{\sqrt[3]{2}}$.

ZAD. 2. Dane są macierze $\mathbf{A} = \begin{bmatrix} 1 & 3 & -8 \\ -10 & 2 & 4 \\ 12 & 1 & 5 \end{bmatrix}$, $\mathbf{B} = \begin{bmatrix} 1 & 2 \\ 3 & 1 \\ -1 & 5 \end{bmatrix}$.

Oblicz, tam gdzie to możliwe

- | | |
|--|--|
| a) wyznaczniki macierzy
b) macierze odwrotne
c) macierze transponowane i ich wyznaczniki | d) iloczyny macierzy (również przez siebie lub macierz transponowaną)
e) iloczyn skalarny pierwszego wiersza macierzy \mathbf{A} oraz drugiej kolumny macierzy \mathbf{B} . |
|--|--|

ZAD. 3. Wykorzystując zapis macierzowy układu równań oraz działania na macierzach, wyznacz macierz odwrotną do macierzy układu $\mathbf{A}^{-1} = [a_{ij}]_{4 \times 4}$ oraz wektor \mathbf{X} rozwiązań układu równań:

$$\begin{cases} x + y - z - 2t = -7 \\ 2x + 2y + 3t = 21 \\ 5x + 5y + 2z + 5t = 40 \\ -2x - 3y + z + 2t = 2 \end{cases} \quad \text{Podaj wartość } a_{23}.$$

ZAD. 4. Skonstruuj wektor kwadratów liczb od 1 do 100 a następnie zlicz, które cyfry oraz jak często występują na pozycji jedności w wyznaczonych kwadratach (użyj operatora *modulo*, funkcji *summary* oraz typu *factor*). Ile wśród nich jest podzielnych przez 2, przez 3 oraz przez 2 i 3?

ZAD. 5. Zbuduj własne tablice trygonometryczne. Przygotuj ramkę danych, w których zebrane będą informacje o wartościach funkcji sinus, cosinus, tangens i cotangens dla kątów: 0° , 30° , 45° , 60° , 90° (funkcje trygonometryczne w R przyjmują argumenty w radianach). W tym celu napisz funkcję **rad** (użyj *function*), która zamieni stopnie na radiany (stała π w R ma oznaczenie *pi*), utwórz wektor w radianach oraz ramkę danych **Tablice** (użyj *data.frame*).

ZAD. 6. Przygotuj wektor 30 łańcuchów znaków następującej postaci: liczba.litera, gdzie liczba to kolejne liczby od 1 do 30 a litera to trzy duże litery A, B, C występujące cyklicznie, czyli 1.A 2.B 3.C 4.A itd. Wykorzystaj funkcję *paste*, która łączy napisy.

Struktury danych w R

Obiekt	Typy	Różne typy
Wektor (<i>vector</i>)	numeryczny, znakowy, zespolony, logiczny	Nie
Czynnik (<i>factor</i>)	numeryczny, znakowy	Nie
Tabela (<i>table</i>)	numeryczny, znakowy, zespolony, logiczny	Nie
Ramka danych (<i>data.frame</i>)	numeryczny, znakowy, zespolony, logiczny	Tak
Lista (<i>list</i>)	numeryczny, znakowy, zespolony, logiczny funkcja, wyrażenie	Tak

Operatory:

- operatory arytmetyczne: +, -, *, /, ^, %% (dodawanie, odejmowanie, mnożenie, dzielenie, potęgowanie, dzielenie modulo);
- operatory logiczne: !, |, & (negacja, alternatywa, koniunkcja);
- operatory relacyjne: >, <, >=, <=, ==, != .

Pełna lista operatorów na stronie:

<https://cran.r-project.org/doc/manuals/r-release/R-lang.html#Operators>

Wybrane (wbudowane) funkcje

Funkcja	Opis
$\log(x)$	Logarytm naturalny z x
$\exp(x)$	Liczba e podniesiona do potęgi x
$\log(x, n)$	Logarytm z x przy podstawie n
\sqrt{x}	Pierwiastek kwadratowy z x
$\text{factorial}(n)$	$n! = 1 \cdot 2 \cdot \dots \cdot n$
$\text{choose}(n, k)$	Symbol Newtona
$\text{abs}(x)$	Wartość bezwzględna z x
$\text{round}(x, \text{digits}=n)$	Zaokrągla x do n miejsc po przecinku
$\sin(x)$	Sinus x (x w rad)
$\cos(x)$	Cosinus x (x w rad)
$\tan(x)$	Tangens x (x w rad)

Wektory w R możemy tworzyć wykorzystując:

- funkcję `c()`, (*combine* – połącz, np. `c(1, 3, 5, 9, 10)`);
- operator `:` tworzący ciąg arytmetyczny o różnicy 1 (np. `1:10`);
- funkcję `seq()`, (*sequence* - sekwencja) tworzy ciąg arytmetyczny (np. `seq(-5,5,by=0.2)`)
- funkcję `rep()`, (*replicate* - powtórz) powtarza elementy zadaną liczbę razy (np. `rep(c(1,2),2)`).

Funkcja	Opis
<code>length(x)</code>	Długość (liczba elementów) wektora x
<code>max(x)</code>	Największa wartość z elementów wektora x
<code>min(x)</code>	Najmniejsza wartość z elementów wektora x
<code>sum(x)</code>	Suma wszystkich wartości wektora x
<code>prod(x)</code>	Iloczyn wszystkich wartości wektora x
<code>sort(x, decreasing = FALSE)</code>	Sortuje (rosnąco) wartości wektora x ; gdy TRUE - malejąco
<code>sample(x, n, replace=TRUE)</code>	Losowanie <i>n</i> elementów wektora x ze zwracaniem (<code>replace=TRUE</code>) lub bez (<code>replace=FALSE</code>)
<code>which(x)</code>	Zwraca te indeksy wektora logicznego x , które mają wartość TRUE , np. <code>which(x == 5)</code> podaje indeksy elementów wektora x równe 5.
<code>unique(x)</code>	Usuwa duplikaty wektora elementów x .
<code>x[i]</code>	Indeksowanie wektora – odwołanie się do elementu wektora o indeksie <i>i</i>
<code>summary(x)</code>	Podsumowanie dla wektora x (minimum, maksimum, średnia, kwantyle) Dla wektora elementów kategoriowych (factor) zlicza elementy na poszczególnych poziomach

Macierze

- `matrix(dane, r, k)` – tworzy macierz o *r* wierszach i *k* kolumnach z *danych*
- `cbind(x1, ..., xn)` - tworzy macierz łącząc podane wektory jako kolumny
- `rbind(x1, ..., xn)` - tworzy macierz łącząc podane wektory jako wiersze

Funkcja	Opis
<code>dim(A)</code>	Wymiar macierzy A
<code>t(A)</code>	Transpozycja macierzy A
<code>sum(A)</code>	Suma elementów macierzy A
<code>det(A)</code>	Wyznacznik macierzy A
<code>A %*% B</code>	Mnożenie macierzy
<code>solve(A)</code>	Tworzenie macierzy odwrotnej
<code>solve(A, B)</code>	Rozwiązanie układu macierzowego AX = B
<code>A[1,2]</code>	Odwołanie się do elementu w pierwszym wierszu i drugiej kolumnie macierzy A