

5. GLOBÁLNÍ A LOKÁLNÍ PRAVIDLA, MATEMATICKÁ ANALÝZA PRO ZAČÁTEČNÍKY I POKROČILÉ

Globální pravidlo: `levá strana=pravá strana`

Globální pravidlo s odloženým vyhodnocením: `levá strana:=pravá strana`

Informace o uložených pravidlech: `?identifikátor`

Zrušení pravidla: `Clear[identifikátor]`

Levá strana libovolného pravidla může obsahovat tzv. vzory. Je-li např. nějaký identifikátor na levé straně pravidla následován podtržítkem, znamená to, že na jeho místě může stát libovolný výraz.

- 1) Při zadání obyčejného pravidla se nejprve vyhodnotí pravá strana, poté se pravidlo uloží.
- 2) Při zadání pravidla s odloženým vyhodnocením se pravidlo ihned uloží, pravá strana se vyhodnocuje až při použití pravidla.

Globální pravidla se aplikují v pořadí od méně obecného k obecnějšímu.

Lokální pravidlo: `levá strana->pravá strana`

Lokální pravidlo s odloženým vyhodnocením: `levá strana:>pravá strana`

Aplikace lokálních pravidel: `výraz /. pravidlo` `výraz /. {pravidlo1,pravidlo2,...}`

Lokální pravidla se aplikují v tom pořadí, ve kterém jsou uvedena (každé pouze jednou).

Použijeme-li `//.` místo `/.` , budou lokální pravidla aplikována opakovaně.

Limita funkce pro $x \rightarrow x_0$: `Limit[f[x],x->x0]`

Jednostranné limity lze počítat pomocí `Direction->"FromAbove"`, resp. `Direction->"FromBelow"`.

Limita funkce více proměnných: `Limit[f[x,y,...],{x,y,...}->{x0,y0,...}]`

Limita posloupnosti pro $n \rightarrow \infty$: `DiscreteLimit[a[n],n->Infinity]`

Taylorův polynom funkce: `Series[f[x],{x,střed,stupeň}]`

Zbytku ve tvaru $O(x^n)$ se lze zbavit použitím `Normal[Series[...]]`.

Globální extrém funkce: `Minimize[f[x,y,...],{x,y,...}]` (analogicky `Maximize`)

Vázaný globální extrém: `Minimize[{f[x,y,...],podmínky},{x,y,...}]`

Lokální extrém numericky: `FindMinimum[f[x,y,...],{{x,x0},{y,y0},...}]` (analogicky `FindMaximum`)

Vázaný lokální extrém numericky: `FindMinimum[{f[x,y,...],podmínky},{{x,x0},{y,y0},...}]`

Řešení diferenciální rovnice pro $x(t)$: `DSolveValue[rovnice,x[t],t]`

Soustava rovnic pro $x(t), y(t), \dots$: `DSolveValue[{rovnice1,rovnice2,...},{x[t],y[t],...},t]`

`DSolveValue` umí najít obecné řešení nebo řešení splňující zadané počáteční podmínky (ty se zapisují také mezi rovnice).

CVIČENÍ

1. Vypočítejte $\lim_{x \rightarrow 0} \frac{\sqrt{\cos 2x+1} - \sqrt{\cos 3x+1}}{\log(x^2+1)}$, zkontrolujte odpověď pomocí grafu.
2. Graf funkce $f(x) = \sin x + \arcsin x$ na intervalu $[-1, 1]$ je překvapivě rovný. Pokuste se to vysvětlit pomocí Taylorova rozvoje této funkce se středem v bodě 0.
3. Jak dobře je funkce kosinus aproximována Taylorovým polynomem stupně 8 se středem v bodě 0? Porovnejte jejich grafy.
4. Najděte přesné polohy globálních a lokálních extrémů funkce $f(x) = 42x^6 - 126x^5 + 105x^4 - 21x^2 + 1$ na intervalu $[-1, 2]$, ověřte výsledek pomocí grafu.
5. Definujme $f(x) = x^{1/x}$, $x > 0$. Spočítejte $\lim_{x \rightarrow 0^+} f(x)$ a $\lim_{x \rightarrow \infty} f(x)$. Ve kterém bodě nabývá f globálního maxima? Ověřte nalezené výsledky pomocí grafu.
6. Najděte řešení diferenciální rovnice $y'(t) = -t \cdot y(t)$ s počáteční podmínkou $y(0) = y_0$, kde $y_0 \in \mathbf{R}$ je parametr. Zakreslete do jednoho obrázku grafy všech řešení na intervalu $[-3, 3]$, která odpovídají hodnotám $y_0 \in \{-2, -1.5, -1, -0.5, 0, 0.5, 1, 1.5, 2\}$.
7. Počtáři ve starověkém Egyptě uměli pracovat se zlomky, měli však symboly pouze pro kmenové zlomky, tj. zlomky s čitatelem 1. Všechna ostatní racionální čísla zapisovali jako součty těchto zlomků, tj. ve tvaru $1/a_1 + 1/a_2 + \dots + 1/a_n$, kde a_1, \dots, a_n jsou navzájem různá přirozená čísla. Naprogramujte funkci `egypt` takovou, že pro zadané racionální číslo $q \in [0, 1]$ najde `egypt [q]` příslušný rozklad a vrátí seznam sčítanců, tj. $\{1/a_1, 1/a_2, \dots, 1/a_n\}$. Použijte tuto funkci k nalezení rozkladu čísla $99999/100000$. O správnosti výsledku se můžete přesvědčit pomocí funkce `Total [seznam]`, která sečte prvky zadaného seznamu.
Návod: Úlohu lze řešit pomocí „hladového“ algoritmu a rekurze. Definujte `egypt [0] = {}`. Je-li $q > 0$, pak při výpočtu `egypt [q]` volte první jmenovatel a_1 tak, aby $1/a_1$ bylo co největší číslo menší než q , tj. $a_1 = \lceil 1/q \rceil$. Rozklad zbytku, tj. čísla $q - 1/a_1$, najděte rekurzivním voláním `egypt [q - 1/a_1]`. K tomuto rozkladu pak připojte dříve získanou hodnotu $1/a_1$.
8. V úloze „hanojské věže“ jsou dány tři kolíky očíslované 1, 2, 3; na prvním z nich je postavena věž z n kotoučů různých velikostí (největší je vespod, nejmenší nahoře), ostatní jsou prázdné. V každém kroku lze přenést jeden kotouč mezi libovolnými dvěma kolíky, a to tak, že větší kotouč nikdy nesmí ležet na menším. Cílem hry je přenést všechny kotouče na třetí kolík. Naprogramujte funkci, která pro zadané n najde posloupnost kroků potřebných k vyřešení úlohy. Např. pro $n = 2$ bude výstupem funkce seznam `{1->2, 1->3, 2->3}` (kde $i \rightarrow j$ značí přenesení jednoho kotouče z kolíku i na kolík j).
Návod: Úlohu lze řešit pomocí rekurze.
9. Definujte v Mathematice funkci

$$f(n, x) = \sqrt{1 + 2\sqrt{1 + 3\sqrt{1 + 4\cdots + n\sqrt{1+x}}}}, \quad n \in \mathbf{N}, x > -1$$

(pro $n = 1$ je $f(1, x) = \sqrt{1+x}$). Experimentálně určete hodnotu výrazu

$$\sqrt{1 + 2\sqrt{1 + 3\sqrt{1 + 4\cdots}}}$$

(je to limita $f(n, 0)$ pro $n \rightarrow \infty$; Mathematica tuto limitu nezvládne, dá se ale uhodnout sestavením seznamu přibližných hodnot $f(n, 0)$).