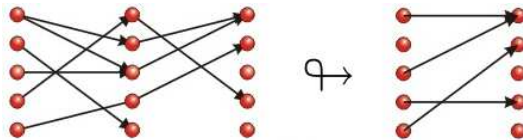


6 Funkce a skládání, Induktivní definice

Vraťme se nyní k látce Lekce 4 z pohledu funkcí a jejich skládání a inverzí.



Kde jste se již intuitivně se skládáním funkcí setkali – jak například spočítáte na kalkulačce výsledek složitějšího vzorce? A jinde?

Na závěr látky o relacích a funkcích se stručně podíváme na problematiku induktivních definic množin a funkcí a uzávěrů vlastností relací. \square

Stručný přehled lekce

- * Přehled základních vlastností funkcí, inverze.
- * Skládání funkcí (coby relací), speciálně aplikováno na permutace.
- * Induktivní definice množin a funkcí; uzávěry vlastností relací.

Relace a funkce, zopakování

- *Relace* mezi množinami A_1, \dots, A_k , pro $k \in \mathbb{N}$, je libovolná podmnožina kartézského součinu

$$R \subseteq A_1 \times A_2 \times \dots \times A_k.$$

Pokud $A_1 = A_2 = \dots = A_k = A$, hovoříme o *k-ární relaci R na A*. \square

- (*Totální*) *funkce* z množiny A do množiny B je relace f mezi A a B taková, že pro každé $x \in A$ existuje *právě jedno* $y \in B$ takové, že $(x, y) \in f$. \square
 - * Množina A se nazývá *definiční obor* a množina B *obor hodnot* funkce f . Funkcím se také říká *zobrazení*.
 - * Místo $(x, y) \in f$ píšeme obvykle $f(x) = y$. Zápís

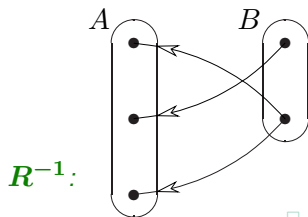
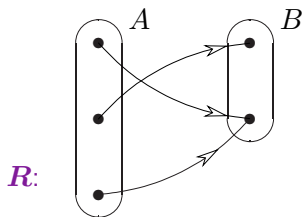
$$f : A \rightarrow B$$

říká, že f je funkce s def. oborem A a oborem hodnot B . \square

- Pokud naší definici funkce upravíme tak, že požadujeme pro každé $x \in A$ *nejvýše jedno* $y \in B$ takové, že $(x, y) \in f$, obdržíme definici *parciální funkce* z A do B .

- *Inverzní relace* k binární relaci R se značí R^{-1} a je definována takto:

$$R^{-1} = \{(b, a) \mid (a, b) \in R\}$$

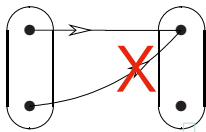


R^{-1} je tedy relace *mezi B a A*.

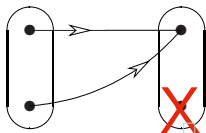
6.1 Vlastnosti funkcí

Definice 6.1. Funkce (případně partiální funkce) $f : A \rightarrow B$ je

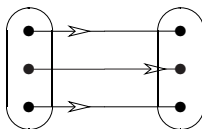
- **injektivní** (nebo také **prostá**) právě když pro každé $x, y \in A$, $x \neq y$ platí $f(x) \neq f(y)$;



- **surjektivní** (nebo také „na“) právě když pro každé $y \in B$ existuje $x \in A$ takové, že $f(x) = y$;



- **bijektivní** (vzáj. **jednoznačná**) právě když je injektivní a **souč.** surjektivní. □



Následují jiné ukázky vlastností funkcí.

- Funkce *plus* : $\mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$ je surjektivní, ale není prostá. \square
- Funkce $g : \mathbb{Z} \rightarrow \mathbb{N}$ daná předpisem

$$g(x) = \begin{cases} -2x - 1 & \text{jestliže } x < 0, \\ 2x & \text{jinak} \end{cases}$$

je bijektivní. \square

- Funkce $\emptyset : \emptyset \rightarrow \emptyset$ je bijektivní. \square
- Funkce $\emptyset : \emptyset \rightarrow \{a, b\}$ je injektivní, ale není surjektivní. \square
- Dokázali byste nalézt bijektivní funkci $\mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$?

Inverze funkce

Příklady inverzí pro relace–funkce.

- Inverzí **bijektivní** funkce $f(x) = x + 1$ na \mathbb{Z} je funkce $f^{-1}(x) = x - 1$. \square
- Inverzí **prosté** funkce $f(x) = e^x$ na \mathbb{R} je **parciální** funkce $f^{-1}(x) = \ln x$. \square
- Funkce $g(x) = x \bmod 3$ **není prostá** na \mathbb{N} , a proto její inverzí je „jen“ relace $g^{-1} = \{(a, b) \mid a = b \bmod 3\}$.
Konkr. $g^{-1} = \{(0, 0), (0, 3), (0, 6), \dots, (1, 1), (1, 4), \dots, (2, 2), (2, 5), \dots\}$.
 \square

Tvrzení 6.2. *Mějme funkci $f : A \rightarrow B$. Pak její inverzní relace f^{-1} je*

- a) **parciální funkce** právě když f je prostá, \square*
- b) **funkce** právě když f je bijektivní.*

Důkaz vyplývá přímo z definic funkce a inverze relace. \square

6.2 Skládání funkcí, permutace

Fakt: Mějme zobrazení (funkce) $f : A \rightarrow B$ a $g : B \rightarrow C$. Pak jejich *složení* $(g \circ f) : A \rightarrow C$ definované

$$(g \circ f)(x) = g(f(x)). \square$$

- Jak například na běžné kalkulačce vypočteme hodnotu funkce $\sin^2 x$? \square
Složíme (v tomto pořadí) „elementární“ funkce $f(x) = \sin x$ a $g(x) = x^2$. \square
- Jak bychom na „elementární“ funkce rozložili aritmetický výraz $2 \log(x^2 + 1)$? \square
Ve správném pořadí složíme funkce $f_1(x) = x^2$, $f_2(x) = x + 1$, $f_3(x) = \log x$ a $f_4(x) = 2x$. \square
- A jak bychom obdobně vyjádřili složením funkcí aritmetický výraz $\sin x + \cos x$?
Opět je odpověď přímočará, vezmeme „elementární“ funkce $g_1(x) = \sin x$ a $g_2(x) = \cos x$, a pak je „složíme“ další funkcí $h(x, y) = x + y$. \square
Vidíme však, že takto pojaté „skládání“ už nezapadá hladce do našeho zjednodušeného formalismu skládání relací.

Skládání permutací

Definice: Necht' *permutace* π množiny $\{1, 2, \dots, n\}$ je určena seřazením jejích prvků (p_1, \dots, p_n) . Pak π je zároveň **bijektivním zobrazením** $\{1, \dots, n\} \rightarrow \{1, \dots, n\}$ definovaným předpisem $\pi(i) = p_i$. \square

Tudíž lze permutace *skládat jako relace* podle Definice 4.7. \square

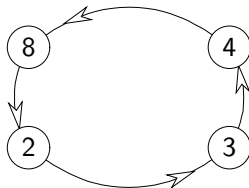
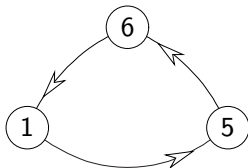
Poznámka: Všechny permutace množiny $\{1, 2, \dots, n\}$ spolu s operací skládání tvoří grupu, zvanou symetrická grupa S_n .

Permutační grupy (podgrupy symetrické grupy) jsou velice důležité v algebře, neboť každá grupa je vlastně isomorfní některé permutační grupě. \square

Příkladem permutace vyskytující se v programátorské praxi je třeba zobrazení $i \mapsto (i+1) \bmod n$ ("inkrement"). \square Často se třeba lze setkat (aniž si to mnohdy uvědomujeme) s permutacemi při indexaci prvků polí.

Cykly permutací

V kontextu pohledu na funkce a jejich skládání coby relací si zavedeme jiný, názornější, způsob zápisu permutací – pomocí jejich **cyklů**.



□

Definice: Necht π je permutace na množině A . **Cyklem** v π rozumíme posloupnost $\langle a_1, a_2, \dots, a_k \rangle$ různých prvků A takovou, že $\pi(a_i) = a_{i+1}$ pro $i = 1, 2, \dots, k-1$ a $\pi(a_k) = a_1$. □

- Jak název napovídá, v zápise cyklu $\langle a_1, a_2, \dots, a_k \rangle$ není důležité, kterým prvkem začneme, ale jen dodržení cyklického pořadí. Cyklus v permutaci může mít i jen jeden prvek (zobrazený na sebe).
- Například permutace $(5, 3, 4, 8, 6, 1, 7, 2)$ je zakreslena svými cykly výše.

Reprezentace permutací jejich cykly

Věta 6.3. Každou permutaci π na konečné množině A lze zapsat jako složení cyklů na disjunktních podmnožinách (rozkladu) A . \square

Důkaz: Vezmeme libovolný prvek $a_1 \in A$ a iterujeme zobrazení $a_2 = \pi(a_1)$, $a_3 = \pi(a_2)$, atd., až se dostaneme „zpět“ k $a_{k+1} = \pi(a_k) = a_1$. Proč tento proces skončí? Protože A je konečná a tudíž ke zopakování některého prvku a_{k+1} musí dojít. Nadto je π prostá, a proto nemůže nastat $\pi(a_k) = a_j$ pro $j > 1$. Takto získáme **první cyklus** $\langle a_1, \dots, a_k \rangle$. \square

Induktivně pokračujeme s hledáním dalších cyklů ve zbylé množině $A' = A \setminus \{a_1, \dots, a_k\}$, dokud nezůstane prázdná. V tomto indukčním kroku si musíme uvědomit, že π omezené na nosnou množinu A' je **stále permutací** podle definice. \square

Značení permutace jejími cykly: Nechť se permutace π podle Věty 6.3 skládá z cyklů $\langle a_1, \dots, a_k \rangle$, $\langle b_1, \dots, b_l \rangle$ až třeba $\langle z_1, \dots, z_m \rangle$. Pak zapíšeme

$$\pi = (\langle a_1, \dots, a_k \rangle \langle b_1, \dots, b_l \rangle \dots \langle z_1, \dots, z_m \rangle).$$

Příklad 6.4. Ukázka skládání permutací daných svými cykly.

Vezměme 7-prvkovou permutaci $\pi = (3, 4, 5, 6, 7, 1, 2)$. Ta se skládá z jediného cyklu $\langle 1, 3, 5, 7, 2, 4, 6 \rangle$. Jiná permutace $\sigma = (5, 3, 4, 2, 6, 1, 7)$ se rozkládá na tři cykly $\langle 1, 5, 6 \rangle$, $\langle 2, 3, 4 \rangle$ a $\langle 7 \rangle$. \square

Nyní určíme složení $\sigma \circ \pi$ těchto dvou permutací (už přímo v zápisu cykly):

$$(\langle 1, 5, 6 \rangle \langle 2, 3, 4 \rangle \langle 7 \rangle) \circ (\langle 1, 3, 5, 7, 2, 4, 6 \rangle) = (\langle 1, 4 \rangle \langle 2 \rangle \langle 3, 6, 5, 7 \rangle)$$

(Nezapomínejme, že první se ve složení aplikuje pravá permutace!) \square

Postup skládání jsme použili následovný:

- * 1 se zobrazí v permutaci vpravo na 3 a pak vlevo na 4. \square
- * Následně 4 se zobrazí na 6 a pak na 1. Tím „uzavřeme“ první cyklus $\langle 1, 4 \rangle$. \square
- * Dále se 2 zobrazí na 4 a pak hned zpět na 2, tj. má samostatný cyklus. \square
- * Zbýlý cyklus $\langle 3, 6, 5, 7 \rangle$ určíme analogicky. \square

6.3 Induktivní definice množin a funkcí

Přímým zobecněním dřívějších rekurentních definic je následující koncept.

Definice 6.5. Induktivní definice množiny.

Jedná se obecně o popis (nějaké) množiny M v následujícím tvaru:

- Je dáno několik pevných (*bázických*) prvků $a_1, a_2, \dots, a_k \in M$. \square
- Je dán soubor *induktivních pravidel* typu

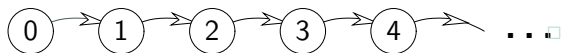
Jsou-li (libovolné prvky) $x_1, \dots, x_\ell \in M$, pak také $y \in M$.

V tomto případě je y typicky funkcí $y = f_i(x_1, \dots, x_\ell)$. \square

Pak naše *induktivně definovaná množina* M je určena jako nejmenší (inkluzí) množina vyhovující těmto pravidlům.

Několik ukázek. . .

- Pro nejbližší příklad induktivní definice se obrátíme na množinu všech přirozených čísel, která je formálně zavedena následovně.
 - $0 \in \mathbb{N}$
 - Je-li $i \in \mathbb{N}$, pak také $i + 1 \in \mathbb{N}$.



- Pro každé $y \in \mathbb{N}$ můžeme definovat jinou množinu $M_y \subseteq \mathbb{N}$ induktivně:
 - $y \in M_y$
 - Jestliže $x \in M_y$ a $x + 1$ je liché, pak $x + 2 \in M_y$. \squarePak například $M_3 = \{3\}$, nebo $M_4 = \{4 + 2i \mid i \in \mathbb{N}\}$. \square
- Dalším příkladem induktivní definice je už známé zavedení **výrokových formulí** z Oddílu 1.5. Uměli byste teď přesně říci, co tam byly základní prvky a jaká byla induktivní pravidla?

Jednoznačnost induktivních definic

Definice: Řekneme, že daná induktivní definice množiny M je *jednoznačná*, právě když každý prvek M lze odvodit z bázeických prvků pomocí induktivních pravidel právě *jedním způsobem*. \square

- Definujme například množinu $M \subseteq \mathbb{N}$ induktivně takto:
 - $2, 3 \in M$
 - Jestliže $x, y \in M$ a $x \leq y$, pak také $x^2 + y^2$ a $x \cdot y$ jsou prvky M .Proč tato induktivní definice není jednoznačná? \square Například číslo $8 \in M$ lze odvodit způsobem $8 = 2 \cdot (2 \cdot 2)$, ale zároveň zcela jinak $8 = 2^2 + 2^2$. \square
- V čem tedy spočívá důležitost jednoznačných induktivních definic množin? Je to především v dalším možném využití induktivní definice množiny jako „základny“ pro odvozené vyšší definice, viz následující.

Definice 6.6. Induktivní definice funkce z induktivní množiny.

Nechť množina M je dána **jednoznačnou** induktivní definicí. Pak říkáme, že funkce $\mathcal{F} : M \rightarrow X$ je definována **induktivně** (vzhledem k induktivní definici M), pokud je řečeno:

- Pro každý z bázičkových prvků $a_1, a_2, \dots, a_k \in M$ je určeno $\mathcal{F}(a_i) = c_i$. \square
- Pro každé induktivní pravidlo typu

“Jsou-li (libovolné prvky) $x_1, \dots, x_\ell \in M$, pak také $f(x_1, \dots, x_\ell) \in M$ ”

je definováno

$\mathcal{F}(f(x_1, \dots, x_\ell))$ na základě hodnot $\mathcal{F}(x_1), \dots, \mathcal{F}(x_\ell)$. \square

Ilustrujme si induktivní definici funkce dětskou hrou na „tichou poštu“. Definičním oborem je řada sedících hráčů, kde ten první je bázičkovým prvkem a každý následující (mimo posledního) odvozuje hráče sedícího hned za ním jako další prvek hry.

Hodnotou bázičkého prvku je první (vymyšlené) posílané slovo. Induktivní pravidlo pak následujícímu hráči přiřazuje slovo, které je odvozeno ze („zkomolením“) slova předchozího hráče. Výsledkem hry pak je hodnota–slovo posledního hráče.

Pro další příklad se podívejme třeba do manuálu unixového příkazu `test`
`EXPRESSION`:

```
EXPRESSION is true or false and sets exit status. It is one of:  
( EXPRESSION )           EXPRESSION is true  
! EXPRESSION             EXPRESSION is false  
EXPRESSION1 -a EXPRESSION2  both EXPRESSION1 and EXPRESSION2 are true  
EXPRESSION1 -o EXPRESSION2  either EXPRESSION1 or EXPRESSION2 is true  
[-n] STRING                the length of STRING is nonzero  
STRING1 = STRING2          the strings are equal  
.....
```

□

No, poněkud problematická je otázka jednoznačnosti této definice – jednoznačnost není vynucena (jen umožněna) syntaktickými pravidly, jinak je pak dána nepsanými konvencemi implementace příkazu.

To je pochopitelně z matematického hlediska velmi špatně, ale přesto jde o pěknou ukázkou z praktického života informatika.

Induktivní definice se „strukturální“ indukci

Příklad 6.7. Jednoduché aritmetické výrazy a jejich význam.

Nechť je dána abeceda $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, \odot, \oplus, (\,)\}$. Definujeme množinu *jednoduchých výrazů* $SExp \subseteq \Sigma^*$ induktivně takto:

- Dekadický zápis každého přirozeného čísla \mathbf{n} je prvek $SExp$. \square
- Jestliže $x, y \in SExp$, pak také $(x) \odot (y)$ a $(x) \oplus (y)$ jsou prvky $SExp$. \square
- Jak vidíme, díky nucenému závorkování je tato induktivní definice jednoduchých výrazů (nikoliv jejich „hodnot“) **jednoznačná**.

Tímto jsme aritmetickým výrazům přiřadili jejich „formu“, tedy *syntaxi*. \square

Pro přiřazení „významu“, tj. *sémantiky* aritmetického výrazu, následně definujeme funkci $Val: SExp \rightarrow \mathbb{N}$ induktivně takto: \square

- Bázické prvky: $Val(\mathbf{n}) = n$, kde \mathbf{n} je dekadický zápis přirozeného čísla n . \square
- První induktivní pravidlo: $Val((x) \oplus (y)) = Val(x) + Val(y)$.
- Druhé induktivní pravidlo: $Val((x) \odot (y)) = Val(x) \cdot Val(y)$. \square

Co je tedy správným významem („**hodnotou**“) uvedených aritmetických výrazů? \square

Příklad 6.8. Důkaz správnosti přiřazeného „významu“ $Val : SExp \rightarrow \mathbb{N}$.

Věta. Pro každý výraz $s \in SExp$ je hodnota $Val(s)$ číselně rovna výsledku vyhodnocení výrazu s podle běžných zvyklostí aritmetiky. \square

Matematickou indukcí: Na rozdíl od dříve probíraných příkladů zde nevidíme žádný celočíselný „parametr n “, a proto si jej budeme muset nejprve definovat. \square

Naši indukci tedy povedeme podle „délky ℓ odvození výrazu s “ definované jako počet aplikací induktivních pravidel potřebných k odvození $s \in SExp$.

\square

Důkaz: V bázi indukce ověříme vyhodnocení základních prvků. Platí $Val(\mathbf{n}) = n$, což skutečně odpovídá zvyklostem aritmetiky. \square

V indukčním kroku se podíváme na vyhodnocení $Val((x) \oplus (y)) = Val(x) + Val(y)$. \square Podle běžných zvyklostí aritmetiky by hodnota $Val((x) \oplus (y))$ měla být rovna součtu vyhodnocení výrazu x , což je podle indukčního předpokladu rovno $Val(x)$ (x má zřejmě kratší délku odvození), a vyhodnocení výrazu y , což je podle indukčního předpokladu rovno $Val(y)$. \square Takže skutečně $Val((x) \oplus (y)) = Val(x) + Val(y)$.

Druhé pravidlo $Val((x) \odot (y))$ se dořeší analogicky. \square

6.4 Uzávěry relací

Definice: Bud' V (nějaká) vlastnost binárních relací. Řekneme, že V je *uzavíratelná*, pokud splňuje následující podmínky:

- Pro každou množinu M a každou relaci $R \subseteq M \times M$ existuje alespoň jedna relace $S \subseteq M \times M$, která má vlastnost V a pro kterou platí $R \subseteq S$.
- Necht' I je množina a necht' $R_i \subseteq M \times M$ je relace mající vlastnost V pro každé $i \in I$. Pak relace $\bigcap_{i \in I} R_i$ má vlastnost V . \square

Fakt: Libovolná kombinace vlastností **reflexivita**, **symetrie**, **tranzitivita** je uzavíratelná vlastnost.

Ireflexivita a antisymetrie **nejsou** uzavíratelné vlastnosti. \square

Věta 6.9. *Necht' V je uzavíratelná vlastnost binárních relací. Bud' M množina a R libovolná binární relace na M . Pak pro množinu všech relací $S \supseteq R$ na M majících vlastnost V existuje **infimum** R_V (vzhledem k množinové inkluzi), které samo má vlastnost V .*

Tuto „nejmenší“ relaci R_V s vlastností V nazýváme *V -uzávěr* relace R .

Tvrzení 6.10. Nechť R je binární relace na M . Pak platí následující poznatky.

- *Reflexivní uzávěr* R je přesně relace $R \cup \{(x, x) \mid x \in M\}$. □

- *Symetrický uzávěr* R je přesně relace

$$\overset{\leftrightarrow}{R} = \{(x, y) \mid (x, y) \in R \text{ nebo } (y, x) \in R\}. \square$$

- *Tranzitivní uzávěr* R je přesně relace $R^+ = \bigcup_{i=1}^{\infty} \mathcal{T}^i(R)$, kde \mathcal{T} je funkce, která pro každou binární relaci S vrátí relaci

$$\mathcal{T}(S) = S \cup \{(x, z) \mid \text{existuje } y \text{ takové, že } (x, y), (y, z) \in S\}$$

a $\mathcal{T}^i = \underbrace{\mathcal{T} \circ \dots \circ \mathcal{T}}_i$ je i -krát iterovaná aplikace funkce \mathcal{T} . □

- Reflexivní a tranzitivní uzávěr R je přesně relace $R^* = Q^+$, kde Q je reflexivní uzávěr R . □
- Reflexivní, symetrický a tranzitivní uzávěr R (tj. nejmenší ekvivalence obsahující R) je přesně relace $(\overset{\leftrightarrow}{Q})^+$, kde Q je reflexivní uzávěr R .
- Na pořadí aplikování uzávěrů vlastností záleží! (Zhruba řečeno, tranzitivní uzávěr aplikujeme coby poslední.)

Nefornálně:

- V -uzávěr relace je vlastně daný induktivní definicí podle vlastnosti V . □
- Význam reflexivních a symetrických uzávěrů je z předchozího zřejmý. □
- Význam tranzitivního uzávěru R^+ je následovný:

Do R^+ přidáme všechny ty dvojice (x, z) takové, že v R se lze „dostat po šipkách“ z x do z . Nakreslete si to na papír pro nějakou jednoduchou relaci, abyste význam tranzitivního uzávěru lépe pochopili. □



- A jak bylo dříve řečeno, antisymetrický uzávěr relace prostě nedává smysl. □
- Například buď $R \subseteq \mathbb{N} \times \mathbb{N}$ definovaná takto: $R = \{(i, i + 1) \mid i \in \mathbb{N}\}$. Pak R^* je běžné lineární uspořádání \leq přirozených čísel.