Vysoké učení technické v Brně

Fakulta informačních technologií

MAT

(tématické okruhy ke státnicím 2009)

Vladimír Veselý – xvesel38

OBSAH

1 Predikátová logika, její jazyk (termy, formule) a sémantika (realizace jazyka, pravdivost formulí)

1.1 Základy logiky

Logikou rozumíme analýzu usuzovacích metod a zkoumání matematických důkazů.

Zobrazení $f: X \to Y$ je relace $f \subseteq X \times Y$:

- $\forall x \in X: \exists y \in Y, y = f(x)$ ⇜⇝ každému x je tedy přiřazeno nějaké y;
- $\forall x \in X, \forall y, z \in Y: y = f(x) \wedge z = f(x) \Rightarrow y = z$ ⇜⇝ zobrazení z X do Y je jednoznačné;

Axiomy jsou výchozí tvrzení dané teorie, nedokazují se, jejich platnost se předpokládá. Z axiomů se dedukcí odvozují další tvrzení, tzv. **důsledky**. Základním požadavkem je **bezespornost** – důsledkem axiomu nesmí být nějaké tvrzení a současně jeho negace. Vedlejším požadavkem je **nezávislost** axiomů, tzn., že žádný axiom není důsledkem zbývajících axiomů.

Matematická tvrzení se zapíší pomocí speciálních znaků – **symbolů** (tvoří abecedu dané teorie). Tvrzení dostanou podobu zvláštních **formulí** – slov sestavených určitým způsobem z daných symbolů (tvoří jazyk teorie). Axiomy jsou zapsány jako formule, které chápeme jako vždy pravdivé. **Odvozovací pravidla** jsou jisté manipulace s formulemi, pomocí nichž a axiomů odvozujeme důsledky.

1.2 Výroková logika

Výroková logika zkoumá způsoby tvorby složených výroků z daných jednoduchých výroků, závislost pravdivosti (resp. nepravdivosti) složeného výroku na pravdivosti výroků, z nichž je složen.

Buď P neprázdná množina symbolů, které nazýváme **prvotní formule**, zpravidla značíme např. písmeny p, q. Tyto hrají úlohu jednoduchých výroků. **Složené výroky** vytváříme z jednoduchých pomocí logických spojek: $\neg$ negace, $\wedge$ (nebo &) konjunkce, $\vee$ disjunkce, $\to$ (nebo $\Rightarrow$) implikace, $\equiv$ (nebo $\Leftrightarrow$ nebo $\leftrightarrow$)ekvivalence.

Symboly jazyka L_P výrokové logiky (nad množinou P) jsou prvky množiny P, logické spojky a závorky (a). Úlohu složených výroků hrají výrokové formule jazyka L_P, definované následovně:

(1) $\forall p \in P$ je výroková formule;
(2) Jsou-li A a B výrokové formule, pak $(\neg A), (A \wedge B), (A \vee B), (A \to B), (A \equiv B)$ jsou výrokové formule;
(3) Každá výroková formule vznikne konečným počtem užití pravidel (1) a (2).

Pravdivostní ohodnocení prvotních formulí je libovolné zobrazení $v: P \to \{0,1\}$, tj. zobrazení, které každé prvotní formuli $p \in P$ přiřadí hodnotu 0 (tj. nepravda) nebo 1 (pravda).

Pravdivostní ohodnocení základních složených výrokových formulí je dáno tabulkou:

$\bar{v}(A)$	$\bar{v}(B)$	$\bar{v}(\neg A)$	$\bar{v}(A \wedge B)$	$\bar{v}(A \vee B)$	$\bar{v}(A \rightarrow B)$	$\bar{v}(A \equiv B)$	$\bar{v}(A \mid B)$	$\bar{v}(A \downarrow B)$
0	0	1	0	0	1	1	1	1
0	1	1	0	1	1	0	1	0
1	0	0	0	1	0	0	1	0
1	1	0	1	1	1	1	0	0

Říkáme, že výroková formule A je **tautologie**, jestliže $v(A) = 1$ pro libovolné ohodnocení A. Jinak řečeno A je pravdivá vždycky, bez ohledu na ohodnocení případných jednotlivých prvotních formulí, ze kterých se skládá, což píšeme $\models A$. Následující výrokové formule jsou tautologiemi:

- $\models (A \vee \neg A)$, což je **zákon vyloučení třetího**;
- $\models (\neg\neg A \equiv A)$, což je **zákon dvojí negace**;
- $\models \neg(A \wedge \neg A)$, což je **vyloučení sporu**.

Říkáme, že výrokové formule A a B jsou **logicky ekvivalentní**, právě když $\bar{v}(A) = \bar{v}(B)$, což znamená $A \equiv B$. Následující formule jsou ekvivalentní:

$$A \equiv B \Leftrightarrow (A \rightarrow B) \wedge (B \rightarrow A)$$
$$A \rightarrow B \Leftrightarrow \neg A \vee B$$
$$A \rightarrow B \Leftrightarrow \neg(A \wedge \neg B)$$
$$A \vee B \Leftrightarrow \neg(\neg A \wedge \neg B)$$
$$A \wedge B \Leftrightarrow \neg(\neg A \vee \neg B)$$
$$A \vee B \Leftrightarrow \neg A \rightarrow B$$
$$A \wedge B \Leftrightarrow \neg(A \rightarrow \neg B)$$

1.3 PREDIKÁTOVÁ LOGIKA

Pro označení libovolného prvku z daného oboru používáme **proměnné** (např. *x, y, z, …*). Mezi prvky z daného oboru mohou být nějaké význačné objekty (0, 1, neutrální prvek), pro něž zavádíme speciální symboly zvané **konstanty**.

K označení operací užíváme **funkcí symboly** (např. *f, g, h, …*). Matematika zkoumá vlastnosti objektů a vztahy mezi nimi. Vlastnosti a vztahy mezi objekty daného oboru, tzv. **predikáty**, (*„být záporným číslem“, „být menším než“, „být prvkem“*) vyjadřujeme pomocí **predikátových symbolů** (např. *p, q, r, …*). Predikát znamená vztah mezi užitým počtem objektů, tedy je příkladem relace. Tím je každému predikátovému symbolu přiřazeno přirozené číslo, jeho četnost, udávající počet jeho argumentů. Je-li četnost rovna n, říkáme, že symbol je n-ární.

Z uvedených symbolů sestavujeme jistým způsobem nejjednodušších tvrzení, vyjádřených tzv. **atomickými formulemi**. Z nich vytváříme složitější formule pomocí logických spojek (stejných jako ve výrokové logice) a pomocí následujících **kvantifikátorů proměnných**:

- **univerzální (obecný) kvantifikátor ∀** vyjadřuje platnost pro všechny objekty daného oboru;
- **existenční kvantifikátor ∃** vyjadřuje existenci požadovaného prvku v daném oboru.

Abecedu predikátové logiky 1. řádu tak tvoří tedy funkční, predikátové a pomocné symboly, proměnné, konstanty, logické spojky a nově i kvantifikátory.

Jazyk predikátové logiky 1. řádu je tedy tvořen:

- **logickými symboly** (proměnné, logické spojky, kvantifikátory, závorky, predikát rovnosti =);
- **speciálními symboly** (funkční symboly s $\mathbb{N}_0^+$-ární četností a predikátové symboly $\mathbb{N}^+$-ární četností).

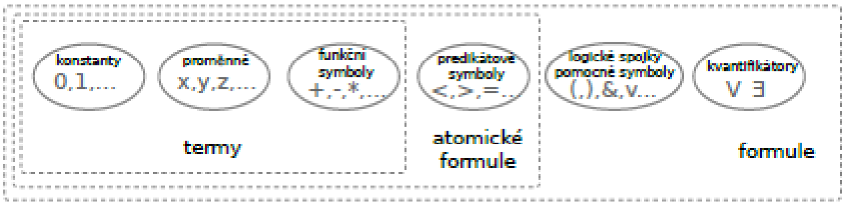
Termy jsou rekurentně definovány následujícími pravidly:

(1) Každá proměnná je term;
(2) Je-li f funkční symbol četnosti n a jsou-li $t_1, \dots, t_n$ termy, pak $f(t_1, \dots, t_n)$ je term;
(3) Každý term vznikne konečným užitím pravidel (1) a (2).

Je-li p predikátový symbol četnosti n a jsou-li $t_1, \dots, t_n$ termy, pak $p(t_1, \dots, t_n)$ nazýváme **atomickou (elementární) formulí**.

Formule je rekurentně definována následujícími pravidly:

(1) Každá atomická formule je formule;
(2) Jsou-li φ, ψ formule, pak také $(\neg\varphi)$, $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \rightarrow \psi)$, $(\varphi \equiv \psi)$ jsou formule;
(3) Je-li x proměnná a φ formule, pak také $(\forall x\varphi)$ a $(\exists x\varphi)$ jsou formule;
(4) Každá formule vznikne konečným užitím pravidel (1), (2) a (3).



Řekneme, že daný **výskyt** proměnné x ve formuli φ **je vázaný**, nachází-li se v nějaké podformuli tvaru $(\forall x\varphi)$ nebo $(\exists x\varphi)$, opačném případě se jedná o **volný výskyt**. V těchto souvislostech hovoříváme o x jako o **vázané/volné proměnné**. Formule neobsahující žádnou volnou proměnnou se nazývá **uzavřená formule** nebo též **výrok**.

1.4 SÉMANTIKA PREDIKÁTOVÉ LOGIKY

Chceme dát interpretaci symbolům jazyka predikátové logiky 1. řádu. Nejprve vymezíme obor, který bude určovat možné hodnoty proměnných, bude to určitý soubor M uvažovaných objektů. Funkčním symbolům budou odpovídat operace na M příslušných četností. Predikátovým symbolům budou odpovídat vztahy mezi objekty z M, které lze popsat jako relace na M s patřičnou aritou. Máme-li jazyk s rovností, interpretujeme symbol = jako rovnost objektů z M.

Nechť L je jazyk 1. řádu, pak **realizací jazyka** L rozumíme algebraickou strukturu $\mathcal{M}$, která se skládá z:

- neprázdné množiny M nazývané **univerzum**;
- pro každý funkční symbol f četnosti n, je dáno zobrazení $f_{\mathcal{M}}: M^n \rightarrow M$;
- pro každý predikátový symbol p četnosti n, krom rovnosti je dána relace $p_{\mathcal{M}} \subseteq M^n$.

Libovolné zobrazení e množiny všech proměnných do univerza M dané realizace $\mathcal{M}$ jazyka L budeme nazývat **ohodnocení proměnných**.

Hodnota termu t v realizaci $\mathcal{M}$ jazyka L při daném ohodnocení e označujeme jako $t[e]$ a indukcí se definuje následovně:

- Je-li t proměnná x, potom $t[e] = e(x)$;
- Je-li t ve tvaru $f(t_1, \dots, t_n)$, kde f je funkční symbol četnosti n a $t_1, \dots, t_n$ jsou termy, potom $t[e] = f_{\mathcal{M}}(t_1[e], \dots, t_n[e])$;

Formule φ **je splněna** v realizaci $\mathcal{M}$ pokud je pravdivá při každém ohodnocení e, píšeme $\mathcal{M} \vDash \varphi$. Je-li φ uzavřená, pak říkáme, že φ **je pravdivá** v $\mathcal{M}$. **Formule** φ jazyka L **je logicky platná**, pokud pro každou realizaci $\mathcal{M}$ jazyka L platí $\mathcal{M} \vDash \varphi$.

Říkáme, že formule φ, ψ jazyka L **jsou logicky ekvivalentní**, jestliže v libovolné realizaci $\mathcal{M}$ jazyka L při libovolné ohodnocení e proměnných, je $\mathcal{M} \vDash \varphi[e] \Leftrightarrow \mathcal{M} \vDash \psi\,[e]$. Každá formule φ jazyka L je logicky ekvivalentní nějaké formuli ψ v níž se nevyskytuje kvantifikátor $\exists/\forall$. Každá formule jazyka L je logicky ekvivalentní nějaké formuli vytvořené z atomických formulí jen pomocí logických spojek $\neg, \rightarrow$ a kvantifikátoru $\forall$. Významné dvojice ekvivalentních formulí:

$$
\begin{aligned}
&(\exists x\varphi) \Leftrightarrow \neg\big(\forall x(\neg\varphi)\big)\\
&(\forall x\varphi) \Leftrightarrow \neg\big(\exists x(\neg\varphi)\big)\\
&(\forall x\varphi) \wedge (\forall x\psi) \Leftrightarrow \forall x(\varphi \wedge \psi)\\
&(\exists x\varphi) \vee (\exists x\psi) \Leftrightarrow \exists x(\varphi \vee \psi)
\end{aligned}
$$

Substituce termů za proměnné: Pokud v termu t dosadíme za proměnné další termy, t zůstává termem. Dosazením termu za proměnné ve formuli vytvoříme opět formuli. Ne vždy je to vhodné, proměnná musí být substituovatelná (proměnná x taková, že žádný její volný výskyt neleží v oboru kvantifikátoru proměnné y, která je obsažena v substituovatelném termu).

2 Axiomy a odvozovací pravidla, dokazatelnost

2.1 Axiomy výrokové logiky

Formální axiomatický systém Hilbertova typu tvoří abeceda (prvotní formule, logické spojky $\neg, \rightarrow$ a závorky) a formule. Jazyk takového formálního axiomatického systému je tvořen abecedou a formulemi. Tři **axiomy výrokové logiky**:

- $A \rightarrow (B \rightarrow A)$;
- $\big(A \rightarrow (B \rightarrow C)\big) \rightarrow \big((A \rightarrow B) \rightarrow (A \rightarrow C)\big)$;
- $(\neg B \rightarrow \neg A) \rightarrow (A \rightarrow B)$.

2.2 Odvozování ve výrokové logice

Jediné odvozovací pravidlo **modus ponens (pravidlo odloučení)**, značí se MP. Z **předpokladů** $A, A \rightarrow B$ lze odvodit **závěr** B.

Formule je dokazatelná, právě když existuje její důkaz! **Důkazem** ve formální výrokové logice rozumíme libovolnou konečnou posloupnost $A_1, \dots, A_n$ výrokových formulí takovou, že pro každé $i \leq n$ formule A_i je buď axiomem nebo je závěrem pravidla modus ponens, jehož předpoklady jsou mezi $A_1, \dots, A_{i-1}$. Řekneme, že formule A je dokazatelná ve výrokové logice, jestliže existuje důkaz, jehož poslední formulí je formule A, což zapisujeme $\vdash A$.

Věta o úplnosti říká, že každá tautologie je dokazatelná. **Postova věta o úplnosti** říká, dokazatelné formule jsou tautologiemi. Obě předchozí věty ukazujíc ekvivalenci $\vdash A \Leftrightarrow \vDash A$. **Věta o korektnosti** říká, že libovolná dokazatelná formule výrokové logiky je tautologií.

Věta o dedukci říká, že je-li T množina formulí, pak $T \vdash A \to B$ ($A \to B$ je dokazatelné pomocí T), právě když $T \cup \{A\} \vdash B$, což píšeme $T, A \vdash B$. Věta o dedukci pomáhá při procesu dokazování.

Neutrální formule neovlivňuje důkaz, platí $T, A \vdash B \wedge T, \neg A \vdash B \Rightarrow T \vdash B$.

2.3 AXIOMY PREDIKÁTOVÉ LOGIKY

Jazyk L predikátové logiky přebíráme z předchozího s tím, že z logických spojek bereme jako základní $\neg, \to$ a jako základní kvantifikátor $\forall$. Analogicky na axiomy výrokové logiky vybudujeme tři základní axiomy predikátové logiky:

- $\varphi \to (\psi \to \varphi)$;
- $\big(\varphi \to (\psi \to \eta)\big) \to \big((\varphi \to \psi) \to (\varphi \to \eta)\big)$;
- $(\neg\psi \to \neg\varphi) \to (\varphi \to \psi)$.

Nově k nim přidáme **schéma axiom kvantifikátoru**, formulemi φ, ψ a proměnnou x, která nemá v φ volný výskyt, pak:

$$\forall x(\varphi \to \psi) \to \big(\varphi \to (\forall x\psi)\big)$$

Dále **schéma axiomu substituce**, ve kterém je-li φ formule, x proměnná a t term substituovatelný za x do φ, pak:

$$\forall x\varphi \to \varphi_x[t]$$

Poslední schéma axiomu rovnosti platí pro predikátové logiky s rovností. Je-li x proměnná, pak $x = x$ je axiom. Jsou-li $x_1, \dots, x_n, y_1, \dots, y_n$ proměnné a je-li f funkční symbol s četností n a p predikátový symbol s četností n, pak jsou axiomy:

- $\Big(x_1 = y_1 \to \Big(x_2 = y_2 \to \big(\dots\big(x_n = y_n \to f(x_1, \dots, x_n) = f(y_1, \dots, y_n)\big) \dots\big)\Big)\Big)$;
- $\Big(x_1 = y_1 \to \Big(x_2 = y_2 \to \big(\dots\big(x_n = y_n \to p(x_1, \dots, x_n) = p(y_1, \dots, y_n)\big) \dots\big)\Big)\Big)$

2.4 ODVOZOVÁNÍ V PREDIKÁTOVÉ LOGICE

Stejně jako ve výrokové logice i zde existuje pravidlo odloučení (modus ponens), kde z formulí $\varphi, \varphi \to \psi$ lze odvodit ψ. Kromě něho zde existuje ještě **pravidlo zobecnění (generalizace)**, kde pro libovolnou proměnnou x z formule φ se odvodí formule $\forall x\varphi$.

Důkazem v predikátové logice 1. řádu rozumíme libovolnou posloupnost $\varphi_1, \dots, \varphi_n$ formulí jazyka L, v níž pro každé i je formule φ_i buď axiom nebo ji lze odvodit z některých předchozích formulí φ_j ,kde $j < i$ použitím pravidla odloučení nebo zobecnění.

Řekneme, že formule φ **je dokazatelná** v predikátové logice 1. řádu, existuje-li důkaz, jehož poslední formulí je φ, což píšeme $\vdash \varphi$. Obecněji, pokud T je množina předpokladů, ze kterých je φ dokazatelná, tak $T \vdash \varphi$.

Věta o korektnosti říká, že libovolná formule jazyka L dokazatelná v predikátové logice 1. řádu, je-li logicky platnou formulí (tj. je splněna v každé realizaci jazyka L).

Jsou-li $x_1, \dots, x_n$ volné proměnné ve formuli φ v nějakém pořadí, pak formuli $(\forall x_1 \dots \forall x_n)$ nazveme **uzávěrem formule** φ.

Je-li L jazyk 1. řádu a T množina formulí jazyka L, říkáme, že T je **teorie 1. řádu** s jazykem L. Říkáme, že teorie T **je sporná**, jestliže pro KAŽDOU formuli φ jazyka L platí $T \vdash \varphi$. Ve sporné teorii totiž lze $T \vdash \psi$ i $T \vdash \neg\psi$, lze dokázat formuli i její negaci. V opačném případě **je** teorie **bezesporná**.

3 MODEL A DŮSLEDEK TEORIE, VĚTY O ÚPLNOSTI A KOMPAKTNOSTI

Buď L jazyk 1. řádu, pak připomeňme, že libovolnou množinu T formulí jazyka L nazýváme teorií 1. řádu s jazykem L. Formule z T jsou tzv. **speciální axiomy**, které spolu s axiomy predikátové logiky tvoří soustavu axiomů teorie T.

Buď T teorie s jazykem L a nechť $\mathcal{M}$ je nějaká realizace jazyka L. Říkáme, že $\mathcal{M}$ je **model teorie** T, jestliže pro $\forall \varphi \in T: \mathcal{M} \vDash \varphi$, což zapisujeme $\mathcal{M} \vDash T$.

Říkáme, že formule φ je **důsledkem teorie** T, jestliže pro každý model $\mathcal{M}$ teorie T je $\mathcal{M} \vDash \varphi$, pak píšeme $T \vDash \varphi$. Důsledek je formule, která je splněna v každém modelu dané teorie.

Má-li teorie T s jazykem L nějaký model, potom je bezesporná.

Gödelova věta o úplnosti říká, že teorie T je bezesporná, právě když má nějaký model.

Teorie T **je úplná**, pokud je bezesporná a pro každou uzavřenou formuli platí buď $T \vDash \psi$ nebo $T \vDash \neg\psi$.

Věta o kompaktnosti říká, že když máme T množinu formulí jazyka L. Pak teorie T má nějaký model, právě když každá její podmnožina $Q \subseteq T$ má taky model.

4 NORMÁLNÍ A PRENEXNÍ TVAR FORMULÍ

4.1 NORMÁLNÍ TVAR

Buď $i_1, \dots, i_n$ libovolná permutace čísel $\{1, \dots, n\}$. Nechť $x_1, \dots, x_n$ jsou proměnné a A je formule predikátové logiky. Pak platí:

- $\vdash (\forall x_1) \dots (\forall x_n) A \leftrightarrow (\forall x_{i_1}) \dots (\forall x_{i_n}) A$;
- $\vdash (\exists x_1) \dots (\exists x_n) A \leftrightarrow (\exists x_{i_1}) \dots (\exists x_{i_n}) A$;

Předchozí věta složitým způsobem říká, že pořadí kvantifikátorů ve formuli lze zaměňovat, aniž by docházelo ke změně významu formule.

Věta o ekvivalenci říká, že když formule A' vznikne z formule A nahrazením některých výskytů podformulí $B_1, \dots, B_n$ po řadě formulemi $B'_1, \dots, B'_n$, pak je-li $\vdash B_i \leftrightarrow B'_i$ pro všechna i = {1, …, n}, pak $\vdash A \leftrightarrow A'$. Jednodušeji, nahrazením podformulí formule A jejich ekvivalentními variantami nedochází ke změně významu původní formule.

4.2 PRENEXNÍ TVAR

Věta o ekvivalenci nás teoreticky vybavila možností upravit formule predikátové logiky podle momentálních potřeb na ekvivalentní tvar, který dává čitelnější a přehlednější zápis nebo ve kterém je rozsah platnosti kvantifikátorů v podformulích buď minimalizován, nebo naopak ve kterém mají všechny kvantifikátory co největší rozsah. Praktickým prostředkem k takovým úpravám jsou následující ekvivalence mezi formulemi, kterým se často zkráceně říká **prenexní operace**, protože se výrazněji uplatňují při převodu formulí do tzv. **prenexní formy**.

Buď z proměnná, která není volná ve formuli A, nechť $\circ$ je některá z výrokových spojek $\wedge$,$\vee$ nebo $\rightarrow$, pak platí:

- $\vdash \forall z(A \circ B) \leftrightarrow (A \circ \forall zB)$
- $\vdash \exists z(A \circ B) \leftrightarrow (A \circ \exists zB)$
- pro implikaci v opačném pořadí $B \rightarrow A$: $\begin{cases} \vdash \forall z(B \rightarrow A) \leftrightarrow (\exists zB \rightarrow A) \\ \vdash \exists z(B \rightarrow A) \leftrightarrow (\forall zB \rightarrow A) \end{cases}$

Nechť A je formule predikátové logiky. Formule A_0 je **variantou** formule A, jestliže vznikne z A postupným nahrazením podformulí tvaru (QxB) podformulemi $(QyB_x[y])$, kde Q je obecný nebo existenční kvantifikátor a y je proměnná nevyskytující se v B ⇝ formule, které „vypadají“ stejně, jen se liší názvem proměnné, jsou ekvivalentními variantami téže formule $\vdash A \leftrightarrow A'$.

Formule A je v **prenexní formě**, jestliže má tvar $Q_1 x_1 \dots Q_n x_n B$, kde:

- $n \geq 0$ a pro každé i = {1, …, n} je Q_i buď $\forall$, nebo $\exists$ kvantifikátor;
- $x_1, \dots, x_n$ jsou navzájem různé proměnné;
- B je otevřená formule (tj. neobsahuje kvantifikátor, všechny jsou totiž vytknuty).

Převedení formule na prenexní tvar:

(1) **Vyloučení zbytečných kvantifikátorů** – Vynecháme všechny ty kvantifikátory Q s proměnnou x v podformulích B, pokud se nevyskytující se volně v B;
(2) **Přejmenování proměnných** – Vyhledáme nejlevější podformuli QxA takovou, že proměnná x se vyskytuje volně v A. Pokud má x výskyt ještě v další formuli výchozí podformule, nahradíme QxA její variantou $Qx'A'$, kde x' je různá od všech jiných proměnných vyskytujících se v převáděné formuli. Tento proces opakujeme do doby, dokud se substitucemi v původní formuli nevyskytují samé unikátní proměnné;
(3) **Eliminace ekvivalence** – provedeme podle předpisu: $A \leftrightarrow B \Rrightarrow (A \rightarrow B) \wedge (B \rightarrow A)$;

(4) **Přesun negace dovnitř** – provádíme postupně náhrady podformulí podle schématu, tak dlouho, dokud se spojka negace nevyskytne bezprostředně před atomickými formulemi:

$$\neg(\forall x A) \Leftrightarrow \exists x \neg A$$
$$\neg(\exists x A) \Leftrightarrow \forall x \neg A$$
$$\neg(A \rightarrow B) \Leftrightarrow A \land \neg B$$
$$\neg(A \lor B) \Leftrightarrow \neg A \land \neg B$$
$$\neg(A \land B) \Leftrightarrow \neg A \lor \neg B$$
$$\neg(\neg A) \Leftrightarrow A$$

(5) **Přesun kvantifikátorů doleva** – pro B, ve kterém se nevyskytuje proměnná x, provádíme náhrady podle schématu, ve kterém je $\bar{Q}$ opačný ku kvantifikátoru Q:

$$(QxA) \lor B \Leftrightarrow Qx(A \lor B)$$
$$(QxA) \land B \Leftrightarrow Qx(A \land B)$$
$$(QxA) \rightarrow B \Leftrightarrow \bar{Q}x(A \rightarrow B)$$
$$B \rightarrow (QxA) \Leftrightarrow Qx(B \rightarrow A)$$

Prenexní forma pro danou formuli není jednoznačná!

Na závěr příklad pro $\forall y\big(\exists x P(x,y) \rightarrow \exists u R(y,u)\big) \rightarrow \forall x S(x,y)$:

(1) se neuplatní, žádné zbytečné proměnné nejsou;
(2) dojde ke dvěma substitucím:
  (a) $\forall y\big(\exists x' P(x',y) \rightarrow \exists u R(y,u)\big) \rightarrow \forall x S(x,y)$
  (b) $\forall y'\big(\exists x' P(x',y') \rightarrow \exists u R(y',u)\big) \rightarrow \forall x S(x,y)$
(3) se neuplatní, neb formule neobsahuje ekvivalenci;
(4) se neuplatní, neb formule neobsahuje negaci, kterou by bylo potřeba zanořit;
(5) proběhne v několika krocích:
  (a) $\forall x \Big(\forall y'\big(\exists x' P(x',y') \rightarrow \exists u R(y',u)\big) \rightarrow S(x,y)\Big)$
  (b) $\forall x \forall y' \Big(\big(\exists x' P(x',y') \rightarrow \exists u R(y',u)\big) \rightarrow S(x,y)\Big)$
  (c) $\forall x \forall y' \Big(\forall x'\big(P(x',y') \rightarrow \exists u R(y',u)\big) \rightarrow S(x,y)\Big)$
  (d) $\forall x \forall y' \exists x' \Big(\big(P(x',y') \rightarrow \exists u R(y',u)\big) \rightarrow S(x,y)\Big)$
  (e) $\forall x \forall y' \exists x' \Big(\exists u\big(P(x',y') \rightarrow R(y',u)\big) \rightarrow S(x,y)\Big)$
  (f) $\underline{\underline{\forall x \forall y' \exists x' \forall u \Big(\big(P(x',y') \rightarrow R(y',u)\big) \rightarrow S(x,y)\Big)}}$

5 UNIVERZÁLNÍ ALGEBRY, PODALGEBRY (GENEROVÁNÍ)

5.1 UNIVERZÁLNÍ ALGEBRY

Buď A množina, $n \in \mathbb{N}_0$, pak zobrazení $\omega: A^n \to A$ nazýváme ***n*-ární operaci** na A. n je četnost (arita) operace.

$$\text{pro } n \in \mathbb{N}_0: \omega: \begin{cases} A^n \to A \\ \underbrace{A \times A \times \dots A}_{n} \to A \\ (x_1, x_2, \dots, x_n) \mapsto \omega x_1 x_2 \dots x_n \end{cases}$$

$$\text{pro } n = 0: \omega: \begin{cases} A^0 = \{\emptyset\} \to A \\ \emptyset \mapsto \omega\emptyset \end{cases} \qquad \text{pro } n = 2: \omega: \begin{cases} A^2 \to A \\ (x, y) \mapsto \omega xy =: x\omega y \end{cases}$$

Buď A množina, $n \in \mathbb{N}_0, D \subseteq A^n$. Potom zobrazení $\omega: D \to A$ se nazývá ***n*-ární parciální operace**. Například dělení je parciální operace na množině $\mathbb{R}$, nelze dělit 0, neexistuje tedy zobrazení s 0 jakožto druhým operandem do $\mathbb{R}$.

Buď A množina, I množina indexů. Pro $i \in I$ buď ω_i n_i-ární operace na A, kde $n_i \in \mathbb{N}_0$. Potom $\mathfrak{A} = (A, (\omega_i)_{i \in I}) = (A, \Omega)$ označuje **univerzální algebru** s nosnou množinou A a souborem operací $(\omega_i)_{i \in I} =: \Omega$

Systém arit operace je soubor $(n_i)_{i \in I}$ se nazývá **typ** algebry (A, Ω). Algebry téhož typu jsou podobné. Například $(\mathbb{Z}, +, -, 0, ., 1)$ je algebra typu (2, 1, 0, 2, 0).

Buď A množina, $\circ$ binární operace na A, pak prvek $e \in A$ se nazývá vzhledem k $\circ$:

- **levý neutrální** $\Leftrightarrow \forall x \in A: e \circ x = x$;
- **pravý neutrální**$\Leftrightarrow \forall x \in A: x \circ e = x$;
- **neutrální**$\Leftrightarrow \forall x \in A: e \circ \mathrm{x} = x \circ e = x$.

Buď A množina, $\circ$ binární operace na A, neutrální prvek e a $x \in A$. Potom prvek $y \in A$ se vůči x nazývá:

- **levým inverzním**$\Leftrightarrow$ $y \circ x = e$;
- **pravým inverzním**$\Leftrightarrow$ $x \circ y = e$;
- **inverzním**$\Leftrightarrow$ $y \circ \mathrm{x} = x \circ y = e$;

Ke každé operaci existuje nejvýše jeden neutrální prvek a inverzní prvek. Ke každé operaci inverzní prvek existovat nemusí (prvek není invertibilní), nebo může být prvek inverzní sám k sobě.

Asociativní zákon: Buď A množina, $\circ$ binární operace na A, pak $\circ$ se nazývá **asociativní** $\Leftrightarrow \forall x, y, z \in A: x \circ (y \circ z) = (x \circ y) \circ z$.

Komutativní zákon: Buď A množina, $\circ$ binární operace na A, pak $\circ$ se nazývá **komutativní** $\Leftrightarrow \forall x, y \in A: x \circ y = y \circ x$.

Distributivní zákon: Pokud jsou $+,\cdot$ binární operace nad A, potom $\cdot$ je **distributivní** nad $+$ $\Leftrightarrow \forall x, y, z \in A$:

- $x \cdot (y + z) = x \cdot y + x \cdot z$
  $\wedge$
- $(y + z) \cdot x = y \cdot x + z \cdot x$

5.2 PODALGEBRY

Buď A množina, pak $\omega: A^n \to A$ je n-ární operace na A $(n \in \mathbb{N}_0), T \subseteq A$. Potom množina T **je uzavřená vzhledem k operaci** $\omega \Leftrightarrow \omega(T^n) \subseteq T$ tj. $t_1, \dots, t_n \in T \Rightarrow \omega t_1 \dots t_n \in T$.

Buď $\mathfrak{A} = (A, (\omega_i)_{i \in I})$ je algebra typu $(n_i)_{i \in I}$ $, T \subseteq A$. Potom se množina T nazývá uzavřená vzhledem k $(\omega_i)_{i \in I} \Leftrightarrow T$ je uzavřená vzhledem k ω_i, pro $\forall i \in I$. V tomto případě se pomocí vztahu $\omega_i^* x_1 \dots x_n =: \omega_i x_1 \dots x_n$,kde $(x_1, \dots, x_n) \in T^{n_i}$ definuje n_i-ární operace ω_i^* na T. Algebra $(T, (\omega_i^*)_{i \in I})$ se pak nazývá **podalgebra algebry** $\mathfrak{A}$. Stručně řečeno je (T, Ω) podalgebrou algebry (A, Ω), když je T uzavřena ke všem operacím z Ω a $T \subseteq A$.

Pokud je (A, Ω) a $\left(T_j\right)_{j \in J}$ je soubor podalgeber, pak jejich průnik $\bigcap_{j \in J} T_j$ je rovněž podalgebra.

Nejmenší podalgebra $\langle S \rangle$ algebry (A, Ω) a $S \subseteq A$, která S obsahuje, je definována jako:

$$\langle S \rangle = \bigcap \{T | T \supseteq S, T \text{ je podalgebrou algebry } (A, \Omega)\}$$

$\langle S \rangle$ se nazývá **podalgebra algebry** (A, Ω) **generovaná množinou** S. Množina S se nazývá **systém generátorů podalgebry** $\langle S \rangle$.

Grupa $(G, \cdot, e, ^{-1})$ je **cyklická** $\Leftrightarrow \exists x \in G: G = \langle x \rangle$. Prvek x se pak nazývá **generátor**.

6 GRUPY, OKRUHY, OBORY INTEGRITY A TĚLESA (POLE)

6.1 KLASIFIKACE ZALOŽENÁ NA GRUPÁCH

Algebra $(A,\cdot)$ typu (2) se nazývá **grupoid**.

Grupoid $(H,\cdot)$ se nazývá **pologrupa** právě tehdy, když je $\cdot$ asociativní.

Pologrupa $(H,\cdot)/(H,\cdot,e)$ se nazývá **monoid** typu (2)/(2, 0), pokud e je neutrální prvek. Monoid je tedy asociativní algebra s neutrálním prvkem.

Monoid $(G,\cdot)/(G,\cdot,e,^{-1})$ se nazývá **grupa** typu (2)/(2, 0, 1)$\Leftrightarrow \forall x \in G$ je invertibilní, tj. $\forall x \in G \exists x^{-1} \in G: xx^{-1} = e$. Grupoid je tedy algebra, která je asociativní, má neutrální prvek a všechny prvky jsou invertibilní. Grupa, která je navíc i komutativní se nazývá **abelovská grupa**.

Algebra $(R,+,\cdot)/(R,+,0,-,\cdot)$ se nazývá **okruh** typu (2, 2)/(2, 0, 1, 2) právě když je to vůči $(R,+)$ abelovksá grupou, vůči $(R,\cdot)$ pologrupa a operace $\cdot$ je distributivní nad $+$. Prvek 0 nazýváme **nulovým prvkem** okruhu vzhledem k $+$.

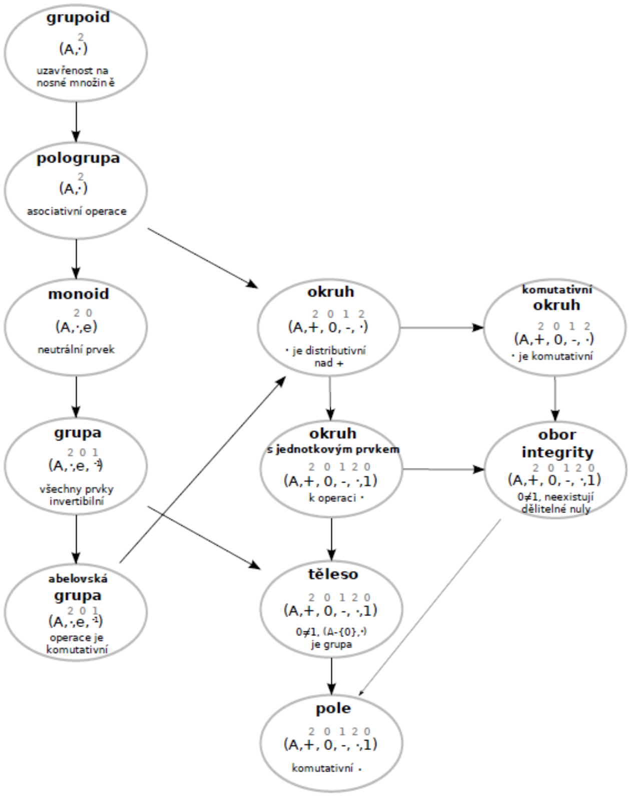
Okruh s jednotkovým prvek je algebra $(R, +, 0, -, \cdot, 1)$ typu $(2, 0, 1, 2, 0)$, kde $(R, +, 0, -, \cdot)$ je okruh a 1 je neutrální prvek vzhledem k $\cdot$, který nazýváme **jednotkovým prvkem** k násobení. Okruh s vlastností komutativity se nazývá **komutativní okruh**.

Komutativní okruh s jednotkovým prvkem se nazývá **obor integrity** $\Leftrightarrow$:

- $R \backslash \{0\} \neq \emptyset$ (tj. $0 \neq 1$ je netriviální, nosná množina obsahuje aspoň dva prvky);
- $\forall x, y \in R: x \neq 0 \wedge y \neq 0 \Rightarrow xy \neq 0$ (tj. neexistují dělitelé 0).

Okruh s jednotkovým prvkem $(R, +, 0, -, \cdot, 1)$ se nazývá **těleso**, pokud je netriviální a neobsahuje dělitele nuly.

Komutativní těleso se nazývá **pole** $\Leftrightarrow$ $0 \neq 1 \wedge (R \backslash \{0\}, \cdot)$ je abelovská grupa. Každé pole je tak obor integrity.



6.2 VLASTNOSTI GRUP

Základní vlastnosti, které vnímáme na množině reálných čísel, jako součin, pravidla pro počítání s mocninami.

Buď $(G, \cdot, e, ^{-1})$ je grupa a $a \in G$. Potom **kardiální číslo (řád prvku)** je množství různých mocnin a:

$$\mathrm{o}(a) = |\{a^0 = e, a^1, a^{-1}, a^2, a^{-2}, \ldots\}| = \left|\{a^k | k \in \mathbb{Z}\}\right|$$

Řádem grupy rozumíme $|G|$ mohutnost nosné množiny, kdy $\forall a \in G: \mathrm{o}(a) \leq |G|$.

6.3 KLASIFIKACE ZALOŽENÁ NA SVAZECH

Svaz je algebra $(V,\cap,\cup)$ typu (2, 2), kde platí, že $\cap,\cup$ jsou komutativní i asociativní a platí **absorpční zákony** $X \cap (X \cup Y) = X$ a $X \cup (X \cap Y) = X$. Obecně nazýváme $\cap$ průsekem a $\cup$ spojením.

U svazů platí **princip duality** kdy svaz je $(V,\cap,\cup)$, právě když $(V,\cup,\cap)$ je svaz.

Svaz je **distributivním svazem**, když platí distributivní zákon svazů, kde je $\cup$ distributivní nad $\cap$, ale i $\cap$ je distributivní nad $\cup$:

$$a \cap (b \cup c) = (a \cap b) \cup (a \cap c) \text{ a } a \cup (b \cap c) = (a \cup b) \cap (a \cup c)$$

Prvek $0 \in V$ se nazývá **nulový prvek svazu V** (neutrální k $\cup$)$\Leftrightarrow \forall a \in V: a \cup 0 = a$ a prvek $1 \in V$ se nazývá **jednotkový prvek svazu V** (neutrální k $\cap$)$\Leftrightarrow \forall a \in V: a \cap 1 = a$. Svaz, který má oba tyto prvky, se nazývá **ohraničený svaz**.

Ohraničený svaz $(V,\cap,\cup,1,0)$ se nazývá **komplementární** $\Leftrightarrow \forall a \in V \exists a' \in V: a \cap a' = 0 \wedge a \cup a' = 1$. Prvek a' se nazývá **komplementem** a.

Distributivní a komplementární ohraničený svaz $(V,\cap,\cup,1,0)$ se nazývá **Booleův svaz**. Algebra $(B,\cap,\cup,1,0,')$ typu (2, 2, 0, 0, 1) se nazývá **Booleova algebra**.

7 HOMOMORFISMY

7.1 RELACE EKVIVALENCE A ROZKLAD NA TŘÍDY EKVIVALENCE

Je-li M množina, potom se podmnožina R množiny $M \times M$ nazývá **binární relace** nad M. Místo $(x,y) \in R$ obvykle píšeme xRy. Relace všech možných dvojic R se nazývá **univerzální relace** $\alpha_M = M \times M$. **Relace rovnosti** nebo také **identická relace** je relace mezi stejným prvkem $\iota_M = \{(x,x) | x \in M\}$. Relace $R \subseteq M \times M$ se nazývá:

- **reflexivní** $\Leftrightarrow \iota_M \subseteq R, tj. \forall x \in M: xRx$;
- **symetrická** $\Leftrightarrow \forall x,y \in M: xRy \Rightarrow yRx$;
- **antisymetrická** $\Leftrightarrow \forall x,y \in M: xRy \wedge yRx \Rightarrow x = y$;
- **tranzitivní** $\Leftrightarrow \forall x,y,z \in M: xRy \wedge yRz \Rightarrow xRz$;

Relace ekvivalence je reflexivní, symetrická a tranzitivní. **Relace (částečného) uspořádání** je reflexivní, antisymetrická a tranzitivní.

Buď M množina. Pak $\mathcal{P} \subseteq \mathfrak{P}(M) = 2^M$ se nazývá **rozklad na třídy ekvivalence**: $\Leftrightarrow$

- $M = \bigcup_{C \in \mathcal{P}} C$;
- $\emptyset \in \mathcal{P}$;
- $A, B \in \mathcal{P}: A = B \vee A \cap B = \emptyset$ tj. každé dvě různé množiny $\mathcal{P}$ jsou vůči sobě disjunktní.

Buď π relace ekvivalence na M, $a \in M$, $[a]_\pi \coloneqq \{b \in M | b\pi a\}$ je tzv. **třída ekvivalence prvku a**. Pak $M/\pi \coloneqq \{[a]_\pi | a \in M\}$ je tzv. **faktorová množina množiny M podle ekvivalence π**.

7.2 HOMOMORFISMY

Při bijekci jsou zobrazením pokryty všechny prvky oboru hodnot (surjekce) a obrazy dvou různých vzoru nejsou stejné (injekce). Z toho rovněž vyplývá stejná mohutnost množin. **Bijekce = injekce + surjekce**.

Buďte $\mathfrak{A} = (A, (\omega_i)_{i \in I})$ a $\mathfrak{A}^* = (A^*, (\omega_i^*)_{i \in I})$ algebry téhož typu $(n_i)_{i \in I}$. Zobrazení $f: A \to A^*$ se nazývá **homomorfismem** algebry $\mathfrak{A}$ do algebry $\mathfrak{A}^*$: $\Leftrightarrow$

- pro $i \in I$, kde $n_i > 0$, platí $\forall x_1, \dots, x_{n_i} \in A: f\left(\omega_i x_1 \dots x_{n_i}\right) = \omega_i^* f(x_1) \dots f\left(x_{n_i}\right)$;
- pro $i \in I$, kde $n_i > 0$, platí $f(\omega_i) = \omega_i^*$.

Homomorfismus zachovává každou operaci, tzn. zobrazení operace ω_i nad prvky z A je to samé, co provedení operace ω_i nad zobrazením jednotlivých prvků. Klasický příkladem je zobrazení logaritmu z algebry $(R, \cdot, 1, ^{-1})$ do algebry $(R, +, 0, -)$.

Uvažujme homomorfismus z předchozí definice, pak existují následující **typy homomorfismů**:

- **izomorfismus** – pokud je f bijektivní (říkáme, že $\mathfrak{A}$ je **izomorfní obraz** $\mathfrak{A}^*$: $\mathfrak{A} \cong \mathfrak{A}^*$);
- **endomorfismus** – zobrazení z algebry do téže algebry $\mathfrak{A} = \mathfrak{A}^*$;
- **automorfismus** – pokud je endomorfní a navíc f je izomorfní;
- **epimorfismus** – pokud f je surjektivní;
- **monomorfismus** – pokud f je injektivní.

8 PŘÍMÉ SOUČINY

Zavádí pravidla pro násobení celých algeber.

Buďte $\mathfrak{A}_k = \left(A_k, \left(\omega_i^k\right)_{i \in I}\right), k \in K$(pole), algebry téhož typu $(n_i)_{i \in I}$ a $A \coloneqq \prod_{k \in K} A_k = \{(a_k)_{k \in K} | a_k \in A_k\}$ je kartézský součin všech množin A_k. Pro všechna $i \in I$ buď operace ω_i na A definována vztahem:

$$\omega_i \left(a_k^1\right)_{k \in K} \dots \left(a_k^{n_i}\right)_{k \in K} \coloneqq \underbrace{\left(\omega_i a_k^1 \dots a_k^{n_i}\right)_{k \in K}}_{\in A_k}$$

U součinu algeber odpovídá nosná množina, kartézskému součinu nosných množin jednotlivých součinitelů a operace jsou definovány pro všechny operace jednotlivých součinitelů.

Algebra $(A_k, (\omega_i)_{i \in I})$ se nazývá **přímý součin** algeber $\mathfrak{A}_k$ a značí se $\prod_{k \in K} \mathfrak{A}_k$.

Přímé součiny pologrup (grup, vektorových prostorů, okruhů, Booleových algeber) jsou opět pologrupy (grupy, vektorové prostory, okruhy, Booleovy algebry).

9 KONGRUENCE A FAKTOROVÉ ALGEBRY

Buďte $\mathfrak{A} = (A, (\omega_i)_{i\in I})$ algebra typu $(n_i)_{i\in I}$ a π relace ekvivalence na A. π se nazývá **relace kongruence** na $\mathfrak{A} \Leftrightarrow \forall i \in I$,kde $n_i > 0$: $a_1, \dots, a_{n_i}, b_1, \dots, b_{n_i}, \in A$ platí:

$$a_1 \pi b_1 \wedge \dots \wedge a_{n_i} \pi b_{n_i} \Rightarrow \omega_i a_1 \dots a_{n_i} \pi \omega_i b_1 \dots b_{n_i}$$

Nad oborem integrity $\mathfrak{A} = (\mathbb{Z}, +, 0, \cdot, 1)$ mějme pevný **modul** $n \in \mathbb{N}_0$ a pro $r, s \in \mathbb{Z}$ je $r \equiv s \bmod n$ (říkáme, že „r **kongruentní s** s **modulo** n“) $\Leftrightarrow n|(r-s)$ (n dělí $r - s$), pak platí:

1) $r \equiv s \bmod n \Leftrightarrow r = s + kn, k \in \mathbb{Z} \Leftrightarrow r, s$ mají stejný zbytek při dělení číslem n;
2) $\equiv \bmod n$ je relace ekvivalence.

Algebru $\mathfrak{A}/\pi \coloneqq (A/\pi, (\omega_i^*)_{i\in I})$ se nazývá **faktorová algebra** algebry $\mathfrak{A}$ podle kongruence π. Často klademe $\omega_i = \omega_i^*$.

10 METRICKÉ PROSTORY, KONVERGENCE

10.1 METRICKÝ PROSTOR

Metrickým prostorem $\mathcal{X} = (X, \rho)$ budeme nazývat libovolnou množinu X prvků, které nazýváme body, pokud na množině X je dána tzv. **vzdálenost**, což je jakákoliv jednoznačná nezáporná reálná funkce $\rho(x, y)$, která je definována pro každou dvojici $(x, y) \in X$ a která splňuje tyto tři podmínky:

1) $\rho(x, y) = 0$, když a jen když $x = y$;
2) $\forall x, y \in X$: $\rho(x, y) = \rho(y, x)$ (symetrie);
3) $\forall x, y, z \in X$: $\rho(x, y) + \rho(y, z) \geq \rho(x, z)$ (trojúhelníková nerovnost).

Příklady metrických prostorů:

- **prostor izolovaných bodů**: $\rho(x, y) = \begin{cases} 0 \text{ když } x = y \\ 1 \text{ když } x \neq y \end{cases}$
- **množina D**^{1}: $\rho(x, y) = |x, y|$ je např. číselná osa oboru hodnot jako $\mathbb{Z}$ či $\mathbb{R}$;
- **množina D**^{n} **uspořádaných** $\boldsymbol{n}$**-tic**: $\rho(x, y) = \sqrt{\sum_{k=1}^{n} (y_k - x_k)^2}$ (Euklidovská metrika).

10.2 KONVERGENCE POSLOUPNOSTÍ, HROMADNÉ BODY

Otevřenou koulí $S(x_0, r)$ v metrickém prostoru $\mathcal{X}$ se středem x_0 a poloměrem r budeme nazývat množinu bodů $x \in \mathcal{X}$, která vyhovuje podmínce:

$$\rho(x, x_0) < r$$

Uzavřenou koulí $S[x_0, r]$ v metrickém prostoru $\mathcal{X}$ budeme nazývat množinu bodů $x \in \mathcal{X}$, která vyhovuje podmínce:

$$\rho(x, x_0) \leq r$$

Otevřenou kouli poloměru ε se středem x_0 budeme nazývat $\boldsymbol{\varepsilon}$**-okolím bodu** x_0 a značit $O_\varepsilon(x_0)$.

Bod x nazýváme **bodem uzávěru množiny** M, jestliže jeho libovolné okolí obsahuje alespoň jeden bod z M. Množina všech bodů uzávěru množiny M se označuje $\overline{M}$ a nazývá **uzávěrem** této množiny.

Protože každý bod, který náleží M, je bodem uzávěru množiny M (tento bod totiž leží v každém svém okolí), platí, že $M \subseteq \overline{M}$. Množinu M, pro kterou platí $M = \overline{M}$, nazýváme **uzavřenou**.

Bod x se nazývá **hromadným bodem množiny** M, jestliže jeho libovolné okolí obsahuje nekonečně mnoho bodů z M.

Nechť $x_1, x_2, \dots$ je posloupnost bodů v metrickém prostoru $\mathcal{X}$. Říkáme, že tato **posloupnost konverguje k bodu** $x \in \mathcal{X}$, jestliže každé ε-okolí $O_\varepsilon(x)$ bodu x obsahuje všechny body x_n počínaje od některého indexu $N(\varepsilon)$, tj. jestliže ke každému číslu $\varepsilon > 0$ lze najít takové číslo $N(\varepsilon)$, že okolí $O_\varepsilon(x)$ obsahuje všechny body x_n, kde $n \geq N(\varepsilon)$. Bod x se nazývá **limita posloupnosti** $\{x_n\}$.

Předchozí definici lze vyslovit také tak, že posloupnost $\{x_n\}$, konverguje k bodu x, jestliže:

$$\lim_{n\to\infty} \rho(x, x_n) = 0$$

11 Banachova věta o pevném bodu

Řadu problémů souvisejících s existencí a jednoznačností řešení rovnic různého typu lze převést na otázku existence a jednoznačnosti pevného bodu nějakého zobrazení odpovídajícího metrického prostoru do tohoto prostoru. Mezi různými kritérii existence a jednoznačnosti pevného bodu zobrazení tohoto druhu můžeme za jedno z nejjednodušších a zároveň nejdůležitějších kritérií považovat tzv. **Banachův princip pevného bodu** (stručně BPPB); někdy též nazývaný **princip kontraktivních zobrazení**.

Nechť $\mathcal{X}$ je metrický prostor. Zobrazení A prostoru $\mathcal{X}$ do prostoru $\mathcal{X}$ se nazývá **kontraktivní** (nebo **kontrakce**), existuje-li takové číslo $\alpha < 1$, že pro libovolné dva body $\forall x, y \in \mathcal{X}$ platí nerovnost: $\rho(Ax, Ay) \leq \alpha\rho(x, y)$.

Bod x se nazývá **pevný bod zobrazení** A, jestliže $Ax = x$. Jinak řečeno, pevné body jsou řešení rovnice $Ax = x$.

Banachova věta o pevném bodu (BPPB) říká, že každé kontraktivní zobrazení definované v úplném metrickém prostoru $\mathcal{X}$ má právě jeden pevný bod.

BPPB lze použít k důkazu vět o existenci a jednoznačnosti řěšení pro rovnice různých typů. Kromě důkazu existence a jednoznačnosti řešení rovnice $Ax = x$ dává BPPB také praktickou metodu přibližného výpočtu tohoto řešení (nazývanou **metoda postupných aproximací**).

12 Normované a unitární prostory

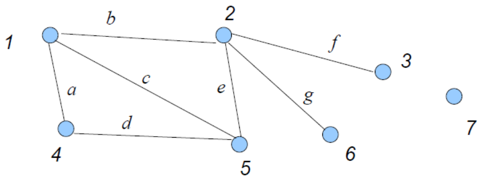
13 Uzavřené ortonormální systémy

14 Fourierovy řady

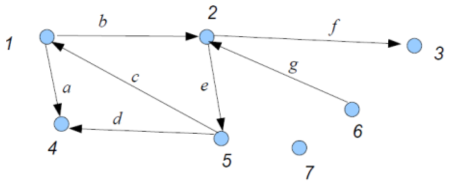
15 OBYČEJNÉ GRAFY, STUPNĚ UZLŮ A JEJICH VZTAH K POČTU HRAN

15.1 OBYČEJNÉ GRAFY A JEJICH VARIANTY

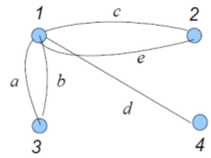
Obyčejný graf je dvojice $G = (U, H)$, kde U je konečná množina uzlů (vrcholů) a $H = \{\{u, v\}: u, v \in U \wedge u \neq v\}$ je konečná množina hran. O hraně h={u,v} říkáme, že je incidentní s uzly u a v, nebo že je mezi uzly u a v, spojuje u a v.



Orientovaný graf je dvojice $G = (U, H)$, kde U je konečná množina uzlů (vrcholů) a $H = \{(u, v): u, v \in U \wedge u \neq v\}$ je konečná množina orientovaných hran.

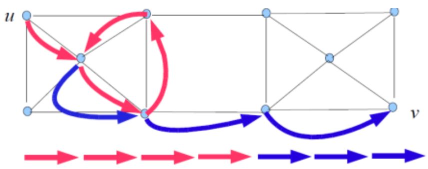


Obecný graf (multigraf) je trojice $G = (U, H, \varepsilon)$, kde U je konečná množina uzlů H je konečná množina hran a ε je zobrazení, které každé dvojici různých uzlů přiřazuje hranu $\varepsilon: \{\{u, v\}: u, v \in U \wedge u \neq v\} \to H$. Mezi jednou dvojicí uzlů tedy může být více hran.

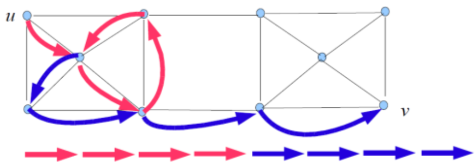


15.2 PRŮCHOD GRAFEM

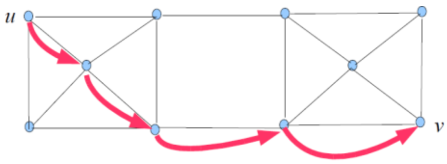
Je-li $G = (U, H)$ obyčejný graf, **sled** mezi uzly u a v o délce n je posloupnost $(u = w_0, h_1, w_1, h_2, \dots, w_{n-1}, h_n, w_n = v)$ takovou, že $w_0, w_1, \dots, w_n \in U$ a kde $h_1, w_2, \dots, h_n \in H$ a že každá hrana spojuje ve sledu dva sousední uzly $h_i = (w_{i-1}, w_i), 1 \leq i \leq n$. Opakovat se můžou uzly i hrany.



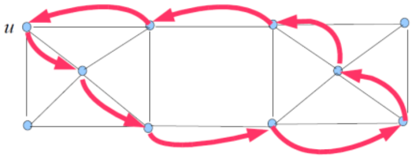
Je-li $G = (U, H)$ obyčejný graf, **tah** mezi uzly u a v o délce n je sled $(u = w_0, h_1, w_1, h_2, \dots, w_{n-1}, h_n, w_n = v)$ takový, že $\forall i, j \in \langle 1, n \rangle: i \neq j \Rightarrow h_i \neq h_j$. V tahu se tedy mohou opakovat uzly, ale už ne hrany.



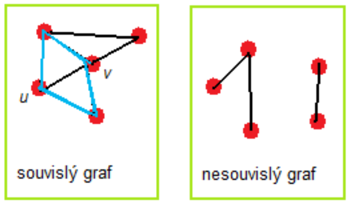
Je-li $G = (U, H)$ obyčejný graf, **cesta** mezi u a v o délce n je sled $(u = w_0, h_1, w_1, h_2, \dots, w_{n-1}, h_n, w_n = v)$ takový, že $\forall i, j \in \langle 1, n \rangle: i \neq j \Rightarrow w_i \neq w_j \wedge h_i \neq h_j$. V cestě se tedy nemohou opakovat ani uzly, ani hrany.



Je-li $G = (U, H)$ obyčejný graf, **kružnice** v grafu G o délce n je sled $(w_0, h_1, w_1, h_2, \dots, w_{n-1}, h_n, w_n)$ takový, že $\forall i, j \in \langle 1, n-1 \rangle: i \neq j \Rightarrow w_i \neq w_j \wedge w_0 = w_n$. Kružnice má všechny hrany a uzly různé s výjimkou 1. a posledního.

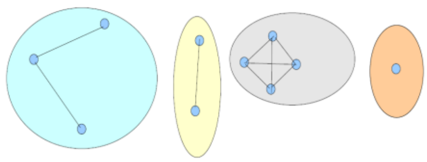


15.3 ČÁSTI GRAFU

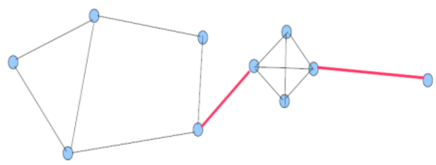


Je-li $G = (U, H)$ obyčejný graf, říkáme, že **je souvislý**, když pro $\forall u, v \in U$ existuje sled $(u = w_0, h_1, w_1, h_2, \dots, w_{n-1}, h_n, w_n = v)$.

Jsou-li $G = (U, H)$ a $G' = (U', H')$ obyčejnými grafy, pak říkáme, že G' **je podgrafem** G, když $U' \subseteq U \wedge H' \subseteq H$. Pokud navíc platí, $(u, v \in U' \wedge \{u, v\} \in H) \Rightarrow \{u, v\} \in H'$ nazývá se podgraf G' **faktorem**. Modrá část v souvislém grafu je podgrafem, není však faktorem, protože chybí hrana spojující u a v.



Jsou-li $G = (U, H)$ a $G' = (U', H')$ obyčejnými grafy, pak říkáme, že G' **je komponentou** G, když G' je souvislým faktorem grafu G a platí: $\left(U' \subset U'' \wedge G'' = (U'', H'') \text{ je podgraf } G\right) \Rightarrow G''$ není souvislý. Komponenta je tedy uzlově maximální souvislý faktor grafu.



Je-li $G = (U, H)$ obyčejný graf a $h \in H$, pak řekneme, že hrana h je **mostem**, pokud by se jejím odstraněním z grafu zvýšil počet komponent grafu. Pokud je hrana $h = \{u, v\}$ mostem, tak jejím odstraněním pak uzly u a v leží v různých komponentách.

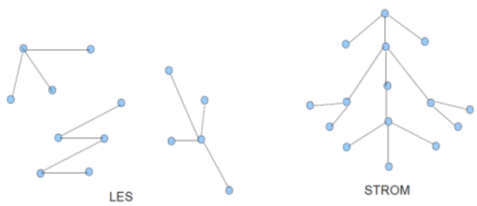
15.4 STUPEŇ UZLU

Je-li $G = (U, H)$ obyčejný graf a $u \in U$, pak definujeme číslo $\deg(u)$ jako **stupeň uzlu**, které nám říká počet hran incidentních s uzlem u.

Nechť je $G = (U, H)$, kde $|H| = m$, pak vztah mezi sumou stupňů všech uzlů a počtem všech hran: $\sum_{u \in U} \deg(u) = 2m$.

16 STROMY A KOSTRY

16.1 STROMY



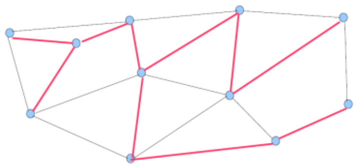
Obyčejný graf, jehož žádný podgraf není kružnicí, se nazývá **les**.

Obyčejný souvislý graf, jehož žádný podgraf není kružnicí, se nazývá **strom**.

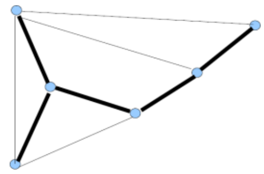
Nechť je $G = (U, H)$ je les, který má aspoň jednu hranu. Pak existují dva uzly $u, v \in U$ takové, že $\deg(u) = \deg(v) = 1$.

Nechť je $G = (U, H)$ je obyčejný graf a $|U| = n, |H| = m$. Pak jsou následující podmínky ekvivalentní G je strom $\Leftrightarrow$ G je souvislý a $m = n - 1$ $\Leftrightarrow$ G neobsahuje jako podgraf kružnici $\Leftrightarrow$ G je souvislý a každá hrana je mostem $\Leftrightarrow$ mezi každou dvojicí různých uzlů v G existuje jediná cesta.

16.2 KOSTRY

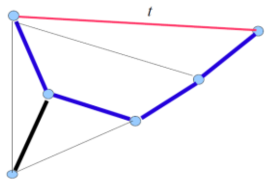


Nechť je $G = (U, H)$ je obyčejný graf, pak jeho podgraf $K = (U, H')$ nazveme **kostrou grafu** G, pokud je K stromem. Každá kostra grafu je tedy uzlově maximální strom obsažený jako podgraf v grafu G.

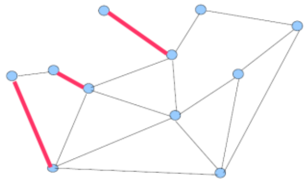


Nechť G je obyčejný graf, pak G je souvislý, právě když má kostru.

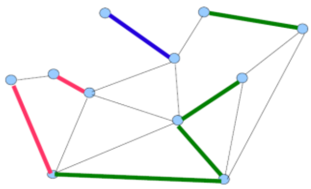
Nechť je $G = (U, H)$ je obyčejný graf a $K = (U, H')$ je jeho kostra. Potom $H' \subseteq H$ a hrany z H, které nejsou v H' se nazývají **tětivy kostry** K.



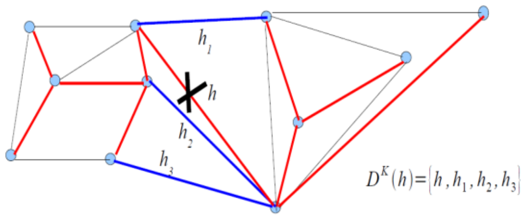
Nechť je $G = (U, H)$ je obyčejný graf a $K = (U, H')$ je jeho kostra a $t = \{u, v\}$ je tětiva kostry K. Pak podle závěrečných ekvivalentních tvrzení, platících pro stromy, existuje jediná cesta mezi uzly u a v v K. Tato cesta pak spolu s tětivou tvoří kružnici v grafu G, kterou nazýváme **základní kružnice kostry** K **vytvořená tětivou** t, což značíme $C^K(t)$.



Nechť je $G = (U, H)$ je obyčejný souvislý graf a $D \subseteq H$ je podmnožina hran. Potom tuto podmnožinu nazveme **rozpojovací množinou grafu** G, pokud odebráním všech hran množiny D z H vznikne nesouvislý graf. Formálně je $G' = (U, H \setminus D)$ nesouvislý graf.



Nechť je $G = (U, H)$ je obyčejný souvislý graf a $D \subseteq H$ je podmnožina hran. Potom tuto množinu nazveme **řezem grafu** G, pokud D je množinově minimální rozpojující množinou grafu G, tedy pokud žádná její vlastní podmnožina není rozpojující množinou. Pokud D je řezem, pak odstraněním se graf rozpadá na komponenty.



Nechť je $G = (U, H)$ je obyčejný graf a $K = (U, H')$ je jeho kostra. Nechť odstraněním hrany h z jeho kostry K, vzniknou dva podgrafy $K_1 = (U_1, H_1)$ a $K_2 = (U_2, H_2)$ a nechť $D = \{h = \{u, v\} | h \in H \wedge u \in U_1 \wedge v \in U_2\}$ je takto vzniklý řez. Potom D nazýváme **základní řez kostry** K **vytvořený hranou** h, což značíme $D^K(h)$.

17 ALGORITMY PRO HLEDÁNÍ MINIMÁLNÍ KOSTRY OHODNOCENÉHO GRAFU

17.1 OHODNOCENÝ GRAF

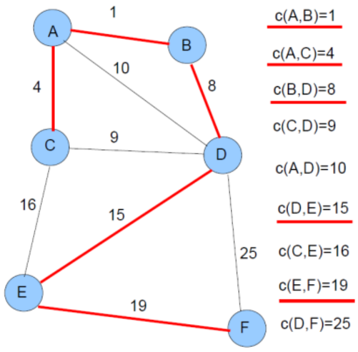
Nechť je $G = (U, H)$ je obyčejný graf. Je-li navíc dáno zobrazení $c: H \to \mathbb{R}$, potom trojici $G = (U, H, c)$ nazýváme **oceněným grafem**. Každé hraně h grafu G je tak přiřazeno reálné číslo $c(h)$, které nazýváme **cenou hrany** h. Je-li $G' = (U', H')$ podgraf grafu G, potom $c'(G') = \sum_{h \in H'} c(h)$ nazýváme **cenou podgrafu** G'.

Nechť $G = (U, H, c)$ je obyčejný oceněný graf a $K = (U, H')$ je kostrou tohoto grafu. Pak říkáme, že K je **minimální kostrou grafu** G, jestliže platí $c(K) \leq c(L)$, kde L je libovolná kostra grafu G.

17.2 ALGORITMY NALEZENÍ MINIMÁLNÍ KOSTRY

Kruskalův algoritmus:

Je dán oceněný obyčejný souvislý graf $G = (U, H, c)$, kde $|U| = n$ a $H = \{h_1, h_2, \dots, h_k\}$. Setřídíme hrany z H do posloupnosti $S = (s_1, s_2, \dots s_k)$,kde $c(s_i) \leq c(s_j)$ pro $i < j$. Odstraníme první hranu $s1$ a vložíme ji do vznikající kostry grafu K. Takto pokračujeme v odstraňování hran z S a vkládáme je do K jen v případě, že by nevznikla v K kružnice, jinak takovou hranu přeskočíme. Algoritmus ukončíme ve chvíli, kdy je K kostrou grafu (obsahuje všechny uzly z U).



Primův algoritmus:

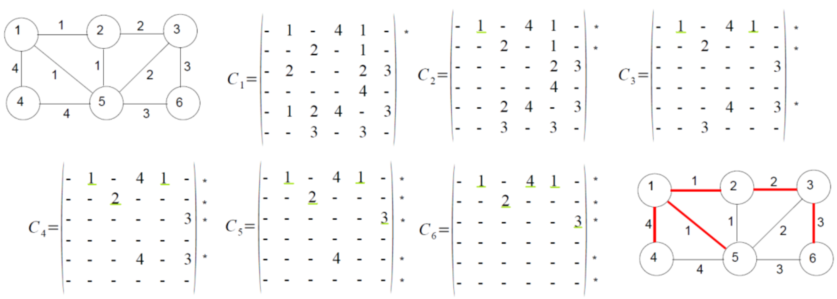
Je dán obyčejný oceněný souvislý graf $G = (U, H, c)$. Pro podgraf $K = (V, J)$ grafu G, který neobsahuje kružnici, označme $K^+ = (V^+, J^+)$ graf, který vznikne z K přidáním uzlu $u \in V$ do V^+ a hrany $h \in J$ do J^+ takové, že:

- h je incidentní s u a s nějakým jiným uzlem z V;
- h nevytvoří v K^+ kružnici;
- h je nejmenší hranou s výše jmenovanými dvěma vlastnostmi.

Primův algoritmus vyjde z libovolného bodu a postupně se přidává vždy hrana s nejmenší cenou taková, že předchozí graf rozšíří tak, aby byl souvislý a přitom neobsahoval kružnici. Oproti Kruskalovu algoritmu má tu výhodu, že se nemusejí předem seřazovat podle vzrůstající ceny všechny hrany. Při Kruskalově algoritmu se totiž většinou hrany s vysokými cenami vůbec nevyužijí.

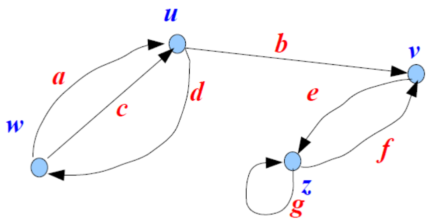
Pokud jsou ceny hran grafu $G = (U, H, c)$ kde $U = \{u_1, u_2, \dots, u_n\}$ zadány ve formě matice, kde prvek na i-tém řádku a v j-tém sloupci označuje hrany incidentní s uzly u_i a u_j. Pak je možno Primův algoritmus vyjádřit v následující formě:

(1) Vyškrtnou se všechny prvky v prvním sloupci a označíme první řádek;
(2) Pokud v označených řádcích neexistuje žádný nepodtržený prvek, algoritmus končí a podtržené prvky označují hrany v minimální kostře. Jinak se vybere minimální prvek;
(3) Je-li vybraný prvek cij, podtrhne se, označí se i-tý řádek a vymažou se nepodtržené prvky j-tého sloupce. Vrátíme se ke kroku (2).



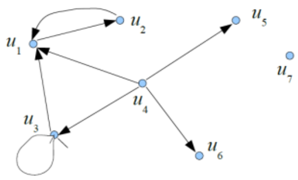
18 ORIENTOVANÉ GRAFY

Orientovaný graf je trojice $G = (U, H, \varepsilon)$, kde U je konečná množina vrcholů, H je konečná množina orientovaných hran. Přitom $\varepsilon: H \to \{(u, v)|u, v \in U\} \cup \{u|u \in U\}$, je zobrazení, které každé hraně přiřadí buď orientovanou dvojici uzlů (u, v) nebo uzel u. V prvním případě říkáme, že hrana vede z uzlu u do v, v druhém případě říkáme, že hrana tvoří smyčku v uzlu u.



Nechť trojice $G = (U, H, \varepsilon)$ je orientovaný graf. Pak definujeme $u \in U$ pro uzel dvě čísla:

- $\deg_+(u) = |\{h \in H | \exists v \in H: \varepsilon(h) = (v, u)\}|$, které nazýváme **vstupním stupněm uzlu**;
- $\deg_-(u) = |\{h \in H | \exists v \in H: \varepsilon(h) = (u, v)\}|$, které nazýváme **výstupním stupněm uzlu**;



Uzel	$\deg_+$	$\deg_-$
u_1	3	1
u_2	1	1
u_3	2	2
u_4	0	4
u_5	1	0
u_6	1	0
u_7	0	0

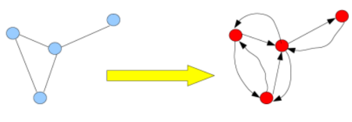
Číslo $\deg_+(u)$ se rovná počtu hran, které vedou z nějakého uzlu do u, číslo $\deg_-(u)$ se rovná počtu hran, které vedou z uzlu u do nějakého uzlu. **Počáteční uzel grafu** má $\deg_+(u) = 0$, **koncový uzel grafu** má $\deg_-(u) = 0$.

Analogicky k obyčejnému grafu, lze definovat varianty orientovaného sledu, orientovaného tahu, orientované cesty a orientované kružnice.

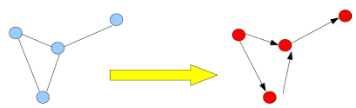
Máme-li zadán obyčejný graf $G = (U, H)$ je k němu možné definovat orientovaný graf $G' = (U, H', \varepsilon)$ tak, že pro každou hranu $\{u, v\} \in H$ existují v H' hrany h_1 a h_2 takové, že $\varepsilon(h_1) = (u, v) \wedge \varepsilon(h_2) = (v, u)$. Takovýto graf G' se nazývá **symetrickou orientací grafu** G. Jinými slovy hrana v obyčejném grafu mezi uzly u a v sebe nahradí oběma orientovanými hranami mezi těmito uzly v novém grafu.

Máme-li zadán obyčejný graf $G = (U, H)$ je k němu možné definovat orientovaný graf $G' = (U, H', \varepsilon)$ tak, že pro každou hranu $\{u, v\} \in H$ existují v H' hrana h taková, že $\varepsilon(h) = (u, v)$ a neexistuje hrana h' taková, že $\varepsilon(h) = (v, u)$. Tento graf se nazývá **orientace grafu** G. Je zřejmé, že na rozdíl od symetrické orientace grafu, která je jednoznačně definována, může existovat k obyčejnému grafu více jeho orientací a navíc orientace grafu neobsahuje smyčky.

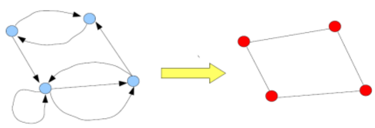
Máme-li zadán orientovaný graf $G = (U, H, \varepsilon)$, potom k němu můžeme sestrojit obyčejný graf $G' = (U, H')$, který se nazývá **symetrizací grafu** G, kde $H' = \{\{u, v\} | u, v \in H, u \neq v, \exists h \in H: \varepsilon(h) = (u, v) \vee \varepsilon(h) = (v, u)\}$. Jinými slovy symetrizace vznikne „zanedbáním" šipek, vícenásobných hran a smyček v původním grafu.



symetrická orientace

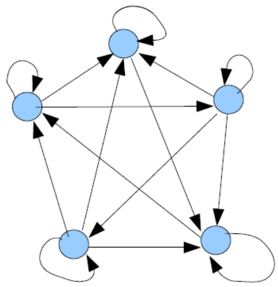


orientace



symetrizace

Říkáme, že orientovaný graf je **slabě souvislý**, jestliže jeho symetrizací vznikne obyčejný souvislý graf. Říkáme, že orientovaný graf $G = (U, H, \varepsilon)$ je **silně souvislý**, jestliže pro libovolné dva uzly $u, v \in U$ existuje orientovaná cesta z u do v.

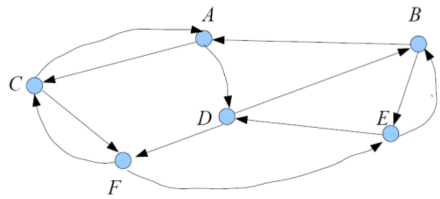


Orientovaný graf $G = (U, H, \varepsilon)$ se nazývá **turnaj**, když pro:

- každou množinu uzlů $\{u, v\}$, že $u, v \in U, u \neq v$, kde existuje právě jedna hrana $h \in H$ taková, že platí $\varepsilon(h) = (u, v) \vee \varepsilon(h) = (v, u)$;
- každý uzel $u \in U$ existuje právě jedna hrana (smyčka) $h \in H$ taková, že platí $\varepsilon(h) = (u, u)$;

Řečeno jinak existuje pro každou dvojici různých uzlů jediná orientovaná hrana jdoucího z jednoho uzlu do druhého a u každého uzlu je smyčka.

19 EULEROVSKÉ GRAFY



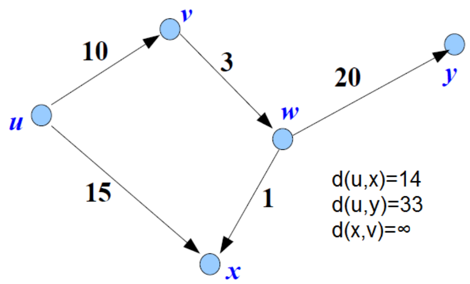
C,A,D,F,C,F,E,B,E,D,B,A,C

Orientovaný graf $G = (U, H, \varepsilon)$ se nazývá **eulerovským grafem**, jestliže v něm existuje UZAVŘENÝ tah (⇜ *„nakreslí se jedním tahem a skončí se tam, kde se začalo“*) délky obsahující všechny orientované hrany. Vzhledem k tomu, že v tahu se nesmějí opakovat hrany, je orientovaný graf eulerovský právě tehdy, když se všechny jeho orientované hrany dají nakreslit ve směru šipek jedním tahem, aniž zvedneme tužku s papíru a po jedné hraně táhneme právě jednou.

Platí věta, že souvislý orientovaný graf $G = (U, H, \varepsilon)$ je právě tehdy eulerovský, když platí $\deg_+(u) = \deg_-(u)$ pro $\forall u \in U$.

20 DÉLKY HRAN A CEST

Pro účely měření délek hran a cest budeme od teď pracovat orientovanými grafy bez vícenásobných hran a smyček. „Hrana“ bude vždy znamenat orientovanou hranu a „cesta“ orientovanou cestu.



Nechť $G = (U, H)$ je graf a každé hraně $h \in H$ nechť je přiřazeno reálné číslo $l(h)$. Potom tomuto číslu budeme říkat **délka hrany** h. **Délka** $d(p)$ **cesty** p v grafu G se definuje jako součet délek všech hran obsažených v cestě p.

Nechť je dán graf $G = (U, H)$ a $u, v \in U$. Pokud existuje mezi uzly u a v cesta minimální délky, definujeme $d(u, v)$ jako délku této cesty. Pokud z uzlu u do uzlu v vůbec žádná cesta neexistuje, klademe $d(u, v) = \infty$.

21 ALGORITMY PRO HLEDÁNÍ CESTY MINIMÁLNÍ DÉLKY

21.1 DIJKSTRŮV ALGORITMUS

Horní odhad vzdálenosti z uzlu s do uzlu v je číslo $D(v)$: $D(v) \geq d(s,v)$.

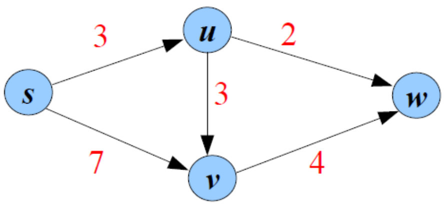
Pro každý uzel $v \in U$ bude symbol $\pi(v)$ označovat uzel, který bezprostředně předchází uzlu v v cestě minimální délky z s do v zkonstruované Dijsktrovým algoritmem. Pokud taková cesta dosud nebyla zkonstruována, pak $\pi(v) = \emptyset$.

Pro každý uzel $v \in U$ definujeme symbol $N(v)$ označující množinu všech uzlů spojených přímo nějakou hranou s uzlem v, tedy $N(v) = \{w \in U | (v,w) \in H\}$.

$S \subseteq U$ je množina všech uzlů v, pro které už byla Dijsktrovým algoritmem definitivně stanovena cesta minimální délky $p(s,v)$ odpovídající minimální vzdálenosti $c(s,v)$.

Schéma Dijsktrova algoritmu po selsku:

(1) Přiřaď vzdálenosti všem uzlům, počátečnímu 0 a všem ostatním nekonečno ∞;
(2) Označ všechny uzly jako nenavštívené, počáteční uzel nastav jako aktuální zpracovávaný;
(3) Pro aktuální uzel zvaž všechny jeho dosud nenavštívené sousedy a přepočítej pro ně vzdálenosti od počátečního uzlu přes aktuální. Pokud je přepočítaná vzdálenost menší, než ta současná, přiřaď mu tuto vzdálenost.
(4) Ve chvíli, kdy je přepočet vzdáleností sousedních uzlů hotov, označ aktuální uzel jako navštívený (už nikdy se nebude kontrolovat, jeho hodnota udává konečnou vzdálenost od počátečního uzlu);
(5) Z množiny dosud nenavštívených uzlů vyber ten s nejmenší vzdáleností od počátečního uzlu a pokračuj krokem (3) do doby, dokud je množina nenavštívených uzlů neprázdná.



1. krok: $S=\{s\}, Q=\{u,v,w\}, D(s)=0, D(u)=D(v)=D(w)=\infty$

2. krok: $S=\{s,u\}, Q=\{v,w\}, D(s)=0, D(u)=3, D(v)=7, D(w)=\infty$
$\pi(u)=s, \pi(v)=s$

3. krok: $S=\{s,u,w\}, Q=\{v\}, D(s)=0, D(u)=3, D(v)=6, D(w)=5$
$\pi(u)=s, \pi(v)=u, \pi(w)=u$

4. krok: $S=\{s,u,w,v\}, Q=\emptyset, D(s)=0, D(u)=3, D(v)=6, D(w)=5$
$\pi(u)=s, \pi(v)=u, \pi(w)=u$

$p(s,u)=s\rightarrow u,\ p(s,v)=s\rightarrow u\rightarrow v,\ p(s,w)=s\rightarrow u\rightarrow w$

Dijkstrův algoritmus nelze použít, pokud se v grafu objevují hrany se zápornou délkou, tento nešvar řeší…

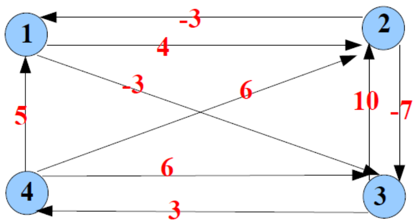
21.2 FLOYD-WARSHALLŮV ALGORITMUS

Při každém zadání délek hran tento algoritmus nalezne cestu minimální délky z každého uzlu do každého jiného uzlu, a pokud taková cesta neexistuje kvůli kružnici se zápornou délkou, tuto kružnici odhalí.

Uvažujme graf $G = (U, H)$, který má $|U| = n$ uzlů a délky hran jsou zadány maticí A, pak budeme používat matici P, kde jsou jednotlivé prvky p_{ij} nastaveny na hodnou sloupce j ve, kterém se nacházejí. Algoritmus má vždy n iterací.

Začneme s maticí $A^0=A$, $P^0=P$ a v i-té iteraci vytvoříme matice A^i, P^i pomocí matic A^{i-1}, P^{i-1}. Nakonec tedy dostaneme matice A^n, P^n. Prvky matic A^j, P^j, $j = 1, 2, \ldots, n$ se vypočítají následujícím způsobem:

- $\texttt{if}\left(a_{ik}^{j-1} \leq a_{ij}^{j-1} + a_{jk}^{j-1}\right)\texttt{then}\left\{a_{ik}^{j} = a_{ik}^{j-1};\ p_{ik}^{j} = p_{ik}^{j-1}\right\}$
- $\texttt{if}\left(a_{ik}^{j-1} > a_{ij}^{j-1} + a_{jk}^{j-1}\right)\texttt{then}\left\{a_{ik}^{j} = a_{ij}^{j-1} + a_{jk}^{j-1};\ p_{ik}^{j} = j\right\}$



$$\Rightarrow A^0 = \begin{bmatrix} 0 & 4 & -3 & \infty \\ -3 & 0 & -7 & \infty \\ \infty & 10 & 0 & 3 \\ 5 & 6 & 6 & 0 \end{bmatrix}, P^0 = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \end{bmatrix}$$

$$j = 1: A^0 = \begin{bmatrix} 0 & 4 & -3 & \infty \\ -3 & 0 & -7 & \infty \\ \infty & 10 & 0 & 3 \\ 5 & 6 & 6 & 0 \end{bmatrix} \Rightarrow A^1 = \begin{bmatrix} 0 & 4 & -3 & \infty \\ -3 & 0 & -7 & \infty \\ \infty & 10 & 0 & 3 \\ 5 & 6 & \underbrace{2}_{6>(5-3)} & 0 \end{bmatrix}, P^1 = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 1 & 2 & 1 & 4 \end{bmatrix}$$

$$j = 2: A^1 = \begin{bmatrix} 0 & 4 & -3 & \infty \\ -3 & 0 & -7 & \infty \\ \infty & 10 & 0 & 3 \\ 5 & 6 & 2 & 0 \end{bmatrix} \Rightarrow A^2 = \begin{bmatrix} 0 & 4 & -3 & \infty \\ -3 & 0 & -7 & \infty \\ \underbrace{7}_{\infty>(-3+10)} & 10 & 0 & 3 \\ \underbrace{3}_{5>(-3+6)} & 6 & \underbrace{-1}_{2>(-7+6)} & 0 \end{bmatrix}, P^2 = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 2 & 2 & 3 & 4 \\ 2 & 2 & 2 & 4 \end{bmatrix}$$

$$j = 3: A^2 = \begin{bmatrix} 0 & 4 & -3 & \infty \\ -3 & 0 & -7 & \infty \\ 7 & 10 & 0 & 3 \\ 3 & 6 & -1 & 0 \end{bmatrix} \Rightarrow A^3 = \begin{bmatrix} 0 & 4 & -3 & \underbrace{0}_{\infty>(-3+3)} \\ -3 & 0 & -7 & \underbrace{-4}_{\infty>(3-7)} \\ 7 & 10 & 0 & 3 \\ 3 & 6 & -1 & 0 \end{bmatrix}, P^3 = \begin{bmatrix} 1 & 2 & 3 & 3 \\ 1 & 2 & 3 & 3 \\ 2 & 2 & 3 & 4 \\ 2 & 2 & 2 & 4 \end{bmatrix}$$

$$j = 4: A^3 = \begin{bmatrix} 0 & 4 & -3 & 0 \\ -3 & 0 & -7 & -4 \\ 7 & 10 & 0 & 3 \\ 3 & 6 & -1 & 0 \end{bmatrix} \Rightarrow A^4 = \begin{bmatrix} 0 & 4 & -3 & \infty \\ -3 & 0 & -7 & -4 \\ \underbrace{6}_{7>(3+3)} & \underbrace{9}_{10>(3+6)} & 0 & 3 \\ 3 & 6 & -1 & 0 \end{bmatrix}, P^4 = \begin{bmatrix} 1 & 2 & 3 & 3 \\ 1 & 2 & 3 & 3 \\ 4 & 4 & 3 & 4 \\ 2 & 2 & 2 & 4 \end{bmatrix}$$

Pro iteraci j sledujeme matici A^{j-1}, a to pouze její j-tý řádek a j-tý sloupec (tedy takový kříž). Pro všechny prvky z A^{j-1} porovnáváme jejich hodnotu s průmětem na tento kříž (tedy se součtem s odpovídajícími prvky na stejném řádku a sloupci v kříži). Pokud je součet menší, v matici A^j zapíšeme součet a v matici P^j zapíšeme hodnotu j.

Pokud kdykoli na hlavní diagonále A vyjde něco jiného než 0, tak v grafu existuje záporná kružnice a tedy neexistuje cesta s minimální délkou.