

Fakulta pedagogická, Technická univerzita v Liberci

DISKRÉTNÍ MATEMATIKA I

Doc. RNDr. Miroslav Koucký, CSc.

Prof. RNDr. Bohdan Zelinka, DrSc.

Liberec, 2004

Obsah

| | | |
|--------------------|---|--------|
| Kap 1. | Základní poznatky o množinách | ... 7 |
| 1.1 | Pojem množiny | ... 7 |
| 1.2 | Relace | ... 17 |
| 1.3 | Zobrazení | ... 25 |
| 1.4 | Operace | ... 28 |
| Kap 2. | Základy matematické logiky | ... 30 |
| 2.1 | Základy výrokové logiky | ... 30 |
| 2.2 | Booleovské funkce | ... 43 |
| 2.3 | Základy predikátové logiky | ... 55 |
| Kap 3. | Základy teorie grafů | ... 58 |
| Přílohy | | |
| • | Použité značení | ... 88 |
| • | Literatura a relevantní zdroje na internetu | ... 90 |

Úvod

Výraz „diskrétní matematika“ zní dosud neobvykle, ačkoliv se ho už nějakou dobu používá. Diskrétní matematiku nelze nějak striktně vymezit, ale můžeme říci, že je opakem „spojité“ matematiky. Objekty, jimiž se zabývá „spojitá“ matematika, tvoří vždy nějaký spojitý celek (kontinuum), ať už jsou to body v geometrii či reálná nebo komplexní čísla. Naproti tomu diskrétní matematika pracuje s objekty od sebe oddělenými a často tvořícími konečné množiny. Zatímco dříve se rozvíjela především matematika „spojitá“ (mající aplikace ve fyzice, v astronomii a v technických vědách), rozvoj počítačů a informatiky přinesl rozvoj matematiky diskrétní. Vždyť například množina čísel v počítači označovaných jako „reálná“ je vlastně konečná; čísla jsou shora i zdola omezena a mají také omezený počet desetinných míst.

Toto je první díl skriptu určeného právě pro studenty informatiky na Pedagogické fakultě a studenty Fakulty mechatroniky Technické univerzity v Liberci. V obou případech jde o první ročník. Skriptum jim má především poskytnout základy těch disciplín, které se zahrnují pod souhrnný název „základy matematiky“, tj. teorie množin a matematické logiky. Dále má seznámit studenty s významným moderním oborem diskrétní matematiky, kterým je teorie grafů.

V první kapitole je popsán pojem množiny a základní množinové operace. Navazuje pojem relace a speciální případy relací jako ekvivalence, uspořádání, zobrazení a operace. Nepředpokládají se žádné předběžné znalosti o množinách a neprovádějí se žádné speciální úvahy o nekonečných množinách. Výklad o matematické logice v druhé kapitole začíná výrokovým počtem a zdůrazňuje se analogie mezi logickými spojkami a množinovými operacemi. Následuje výklad o booleovských funkcích, které jsou v podstatě aplikací matematické logiky v informatice a kapitola končí vyložení základů predikátového počtu. Třetí kapitola je věnována teorii grafů. Omezuje se na konečné neorientované grafy a pouze se zmiňuje o existenci orientovaných grafů. I tato část se ovšem musí omezit na nejzákladnější fakta o základních pojmech této teorie. Jako ukázky aplikací se uvádí Borůvkův problém minimální kostry grafu a čínský problém listonoše.

Na rozdíl od matematiky, na kterou byli čtenáři zvyklí ze základní a ze střední školy, studium matematiky (a nejen diskrétní) na vysoké škole neznamená jen pohotové řešení úloh, ale i zvládnutí teorie, která zahrnuje i určitou faktografii. Té je nutno se naučit a umět ji pak

srozumitelně vyložit tak, aby ten, kdo o věci nic neví, pochopil, o co jde. To dělá studentům často potíže. Doufejme, že i toto skriptum pomůže k jejich překonání.

Přes usilovnou snahu autorů lze očekávat, že jako každá skripta i tato obsahují celou řadu nepřesností, resp. chyb. Autoři budou vděční za každou připomínku vedoucí ke zlepšení věcné i formální stránky následujícího textu[†].

Za technickou práci spojenou s realizací tohoto skripta děkujeme paní Lence Němcové, sekretářce katedry aplikované matematiky TUL.

Autoři

[†] Připomínky lze posílat elektronicky na adresu autorů: miroslav.koucky@tul.cz Opravy textu lze nalézt na URL: <http://www.fp.vslib.cz>, odkaz Učební texty & informace → KAP → Diskrétní matematika.

Kapitola 1. Základní poznatky o množinách

1.1 Pojem množiny

Proč množiny

Na pojmu množiny je založena prakticky celá moderní matematika. Proto se mluví o tzv. základech matematiky, které se skládají z teorie množin a z matematické logiky (o ní zde budeme také mluvit). Slovo „základy“ vůbec neznamená, že by šlo o to nejjednodušší, co v matematice je. Právě naopak, tím slovem je míněno to, že zmíněné dva obory představují obdobu základů stavby. Stojí na nich celá budova matematiky a aby se nezhroutila, musí být důkladné. A skutečně, teorie množin a matematická logika patří k nejsložitějším matematickým oborům.

Tenhle úvod asi čtenáře trochu vystrašil. V tomto skriptu to nebude tak zlé. Pro potřeby diskrétní matematiky se nepotřebujeme důkladně seznamovat s tou opravdovou teorií množin, založenou na axiomatickém základě (axiomatika Bernaysova-Gödelova či Zermelova-Fraenkelova). Postačí nám tzv. naivní teorie množin, založená spíše na názoru.

Co je to množina

Ted' by možná čtenář čekal definici množiny. A přesto tu definice není. Pojem množiny totiž patří mezi základní matematické objekty, které se nedefinují. Definujeme-li totiž nějaký pojem, používáme při tom nějakých dalších pojmů. Ty by pak měly být také definovány pomocí dalších pojmů. Takhle ovšem nelze jít do nekonečna. A nelze jít ani do kruhu. Pokud bychom pojem B definovali pomocí pojmu A , pojem C pomocí pojmu B a pojem A pomocí pojmu C , nedověděli bychom se, ani co je A , ani co je B , ani co je C . Dopustili bychom se tak chyby v logickém uvažování, které se říká definice kruhem. Proto se u určitých pojmů musíme při definování zastavit a nechat je už nedefinované. To jsou tzv. základní pojmy. Patří k nim číslo, bod, přímka, rovina, a také množina. Nevyslovíme tedy žádnou definici, jen si množinu intuitivně představíme jako souhrn nějakých objektů zvaných prvky množiny, což mohou být jakékoliv objekty. Důležité při tom je, abychom u libovolného objektu mohli přesně určit, zda do dané množiny patří či nikoliv. Vyhnete se tedy vágním označením jako „množina veselých lidí“ či „množina krátkých úseček“. Můžeme však mít například množinu obcí v okrese Liberec, množinu občanů České republiky, množinu studentů prvního ročníku TUL a ovšem množiny souvisící s matematikou. Množiny značíme

zpravidla velkými tiskacími písmeny latinské abecedy (např. A, B, C, M, S), případně s indexy a jinými danými symboly (např. $A_0, B_1, C_2, D', E^*, \tilde{F}$). Pro určité množiny čísel používáme standardní označení. Množinu přirozených čísel značíme N (číslo nulu počítáme mezi přirozená čísla), množinu celých čísel Z , množinu racionálních čísel Q , množinu reálných čísel R a množinu komplexních čísel C .

Skutečnost, že x je prvkem množiny M , zapisujeme $x \in M$ a říkáme tomu bodová inkluze. (Prvky množiny často také označujeme jako body množiny, i když s body v geometrickém smyslu nemívají nic společného.) Negací toho tvrzení, tedy skutečnost, že x není prvkem množiny M , zapisujeme $x \notin M$, tedy škrtneme symbol \in (podobně jako se při počítání s čísly užívá škrtnutého rovnítka \neq). Někdy píšeme také $M \ni x$ a čteme to, že množina M obsahuje prvek x .

Symbol \in je jednou z podob, jakými se píše malé řecké písmeno epsilon. V podobě ε se užívá stejně jako většina jiných písmen k označování čísel. Podoba ϵ je vyhrazena jako symbol bodové inkluze.

Množina může obsahovat konečně mnoho nebo nekonečně mnoho prvků. Obsahuje-li množina M konečně mnoho prvků, pak se jejich počet nazývá mohutnost množiny M a značí se $|M|$. Speciálně, existuje množina mohutnosti 0, tedy neobsahující žádný prvek. Taková množina se nazývá prázdná množina a označujeme ji speciálním symbolem \emptyset , resp. $\{\}$. Množina, která není prázdná, se nazývá neprázdná.

Výše uvedený pojem mohutnosti lze zavést i pro množiny s nekonečně mnoha prvky (tzv. nekonečné množiny). V této souvislosti se omezíme pouze na následující konstatování:

- Číselné množiny $N = \{0, 1, 2, \dots\}$, $Z = \{\dots, -2, -1, 0, 1, 2, \dots\}$, $Q = \left\{ \frac{p}{q} \mid p, q \in Z, q \neq 0 \right\}$ (tj. přirozená, celá, racionální čísla) jsou nekonečné, jejich mohutnost označujeme \aleph_0 (čteme alef nula; alef je první písmeno hebrejské abecedy) a říkáme, že jsou spočetné.
- Číselné množiny R, C (reálná, komplexní čísla) jsou také nekonečné, ovšem jejich mohutnost označujeme \aleph_1 (tedy nerovná se \aleph_0) a říkáme, že jsou nespočetné.

(Podrobněji viz odstavec 1.3 Zobrazení)

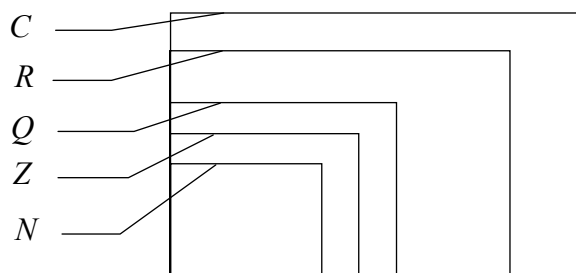
Podmnožiny a nadmnožiny dané množiny

Mějme dvě množiny A, B . Řekneme, že množina B je podmnožinou množiny A , píšeme $B \subseteq A$, jestliže každý prvek množiny B je také prvkem množiny A . Takto definovanému vztahu se říká množinová inkluze (na rozdíl od již popsané bodové inkluze).

Poznámka

V případě, kdy $B \subseteq A$ a $A \neq B$ (tj. existuje prvek z množiny A , který není v B) říkáme, že B je vlastní podmnožina množiny A , a píšeme $B \subset A$.

Například množina přirozených čísel N je podmnožinou celých čísel Z , množina racionálních čísel Q je podmnožinou reálných čísel R a reálná čísla jsou podmnožinou komplexních čísel C . Dostáváme tak vztah $N \subset Z \subset Q \subset R \subset C$ (skutečně jde o vlastní podmnožiny). Graficky lze vztahy mezi uvedenými číselnými množinami (číselnými obory) znázornit následovně:



Následující tvrzení popisují některé základní a zřejmé vlastnosti množinové inkluze:

- Pro každou množinu A platí

$$\emptyset \subseteq A,$$

tedy prázdná množina je podmnožinou libovolné množiny. Dále

$$A \subseteq A, \quad (\text{reflexivita})$$

tj. každá množina je podmnožinou sama sebe.

- Pro každé dvě množiny A, B platí:

$$\text{Je-li } B \subseteq A \text{ a } A \subseteq B, \text{ potom } A = B. \quad (\text{antisymetrie})$$

(Jde vlastně o definici rovnosti množin A, B .)

- Pro každé tři množiny A, B, C platí:

$$\text{Je-li } A \subseteq B \text{ a } B \subseteq C, \text{ potom } A \subseteq C. \quad (\text{tranzitivita})$$

Poznámka

Skutečnost, že množina B je podmnožinou A , můžeme vyjádřit i slovy „ A je nadmnožinou B “ a zápisem $A \supseteq B$. Pojem vlastní nadmnožina a symbol \supset zavádíme analogicky. Například množina racionálních čísel je (vlastní) nadmnožinou celých čísel, současně však je podmnožinou reálných čísel.

Různé způsoby zadávání množin

Zadat množinu znamená jednoznačně určit prvky, které ji tvoří. Pro naše potřeby vystačíme s následujícími třemi způsoby zadávání množin:

1. Zadání množiny výčtem všech jejích prvků

Má-li množina konečný (a dostatečně malý) počet prvků, můžeme ji zadat výčtem všech jejích prvků uvedených ve složených závorkách.

Například množina prvočísel menších než 20 se dá zapsat $\{2, 3, 5, 7, 11, 13, 17, 19\}$. Prvky se mohou zapisovat i slovy, např. množina všech evropských monarchií je množina $\{\text{Belgie, Británie, Dánsko, Lichtenštejnsko, Lucembursko, Monako, Nizozemsko, Norsko, Španělsko, Švédsko, Vatikán}\}$.

Poznamenejme, že velice často se tento způsob zadávání užívá u jednoprvkových množin, např. $\{a\}$ a někdy i u prázdné množiny (v tomto případě píšeme $\{\}$, tj. složené závorky a v nich nic).

2. Zadání množiny pomocí predikátu náležení

Mějme množinu M a její podmnožinu skládající se právě ze všech prvků x množiny M , které mají určitou vlastnost $P(x)$. Potom tuto podmnožinu zapíšeme $\{x \in M \mid P(x)\}$.

Pokud je zřejmý obor M (univerzum), ze kterého prvky vybíráme, lze použít zjednodušený zápis $\{x \mid P(x)\}$.

Například zápisem $\{x \in \mathbb{N} \mid 10 \leq x < 2000\}$ označíme množinu všech přirozených čísel, která jsou větší nebo rovna 10 a menší než 2000. Pokud označíme \check{C} množinu všech Čechů, kteří žili ve 20. století (není důvod, proč bychom množinu nemohli značit i s užitím českého háčku), pak $\{x \in \check{C} \mid x \text{ získal Nobelovu cenu}\} = \{\text{Heyrovský, Seifert}\}$.

3. Rekurzivní zadání množiny

Tento způsob zadávání se využívá především pro tzv. nejvýše spočetné množiny (tj. konečné nebo spočetné) a spočívá v zadání konečného počtu prvků této množiny a jednoznačného předpisu, jak z těchto prvků postupně „generovat“ všechny ostatní prvky.

Například množinu lichých přirozených čísel L lze zadat následovně:

- $1 \in L$,
- jestliže $n \in L$, potom $n + 2 \in L$.

Poznamenejme, že prvkem množiny může být téměř cokoliv, tedy třeba také množina. Můžeme tak mít i množinu množin (nikoliv však množinu všech množin). Aby to znělo trochu lépe, říkáme raději soustava množin nebo systém množin. Pokud píšeme prvky této soustavy (jakožto množiny) velkými tiskacími písmeny, označíme soustavu trochu odlišně, například velkým psacím písmenem, např. $M \in \mathfrak{S}$, což znamená, že množina M je prvkem soustavy množin \mathfrak{S} .

Základní operace s množinami

Jako se provádějí různé početní úkony čili operace (z latinského slova „opus“ čili „dílo“) s čísly, provádějí se i různé operace s množinami. Mezi základní patří průnik, sjednocení a rozdíl množin.

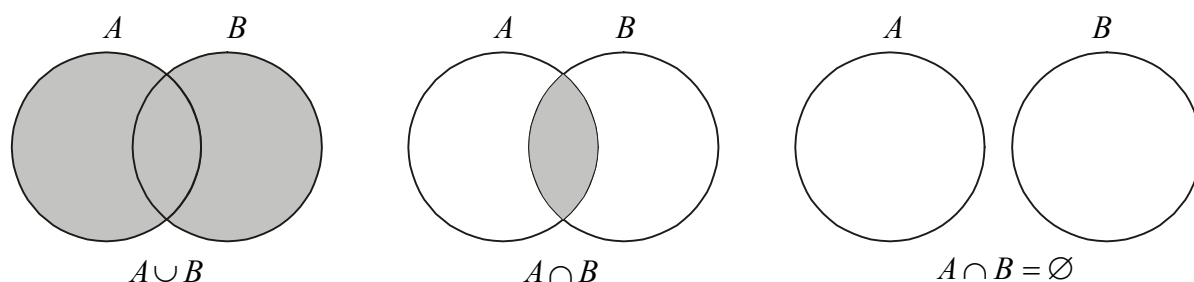
Definice (sjednocení množin, průnik množin, disjunktní množiny)

Nechť A, B jsou dvě libovolné množiny.

- Sjednocení množin A a B označujeme $A \cup B$ a definujeme jako množinu všech prvků, které jsou obsaženy alespoň v jedné z množin A, B , tj. $A \cup B = \{x \mid x \in A \text{ nebo } x \in B\}$.
- Průnik množin A, B označujeme $A \cap B$ a definujeme jako množinu všech prvků, které jsou prvky obou množin A, B současně, tj. $A \cap B = \{x \mid x \in A \text{ a } x \in B\}$.
- Řekneme, že množiny A, B jsou disjunktní, jestliže jejichž průnik je prázdná množina, tj. $A \cap B = \emptyset$.

Operace s množinami, resp. vztahy mezi nimi lze velmi názorně zobrazit pomocí tzv.

Vennových diagramů (John Venn, 1834 - 1923). Pro výše zavedené pojmy dostáváme:



Příklad

Nechť A je množina všech asijských států a E množina všech evropských států. Potom $A \cup E$ je množina všech států Asie a Evropy dohromady, zatímco $A \cap E = \{\text{Rusko, Turecko}\}$. Zde je průnik neprázdný. Kdybychom místo A vzali množinu A_f všech afrických států, bylo by $A_f \cap E = \emptyset$ (žádný stát není současně africký a evropský).

Poznámka (disjunktnost více než dvou množin)

Pojem disjunktnosti je používán i v případě více než dvou množin. Ovšem v této situaci je třeba rozlišovat mezi disjunktností (někdy se mluví o sdružené disjunktnosti) a disjunktnosti po dvou. Rozdíl mezi oběma pojmy je dán následující definicí:

Řekneme, že množiny A_1, \dots, A_m jsou (sdruženě) disjunktní, jestliže

$$A_1 \cap A_2 \cap \dots \cap A_m = \emptyset.$$

(Tedy neexistuje prvek společný všem množinám.)

Řekneme, že množiny A_1, \dots, A_m jsou disjunktní po dvou, jestliže

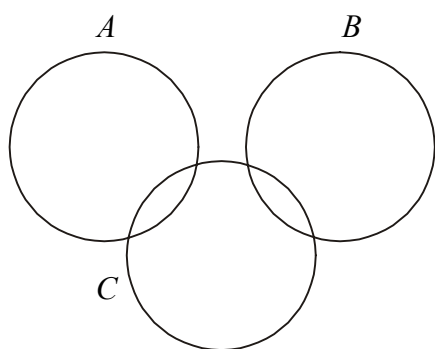
$$\forall i \neq j \quad A_i \cap A_j = \emptyset.$$

(Tedy každá dvojice množin je disjunktní.)

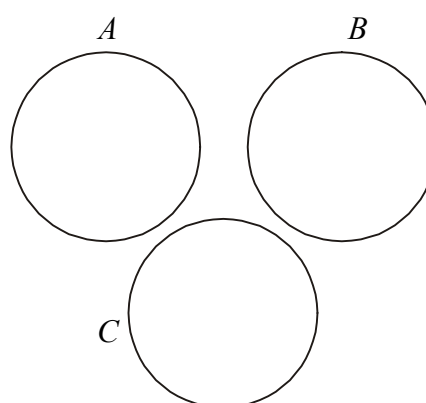
Příklad

- Množiny $A = \{a, b, c\}$, $B = \{1, 2, 3, 4\}$, $C = \{\otimes, a, \oplus\}$ jsou disjunktní, ale nejsou disjunktní po dvou ($A \cap C = \{a\}$).
- Množiny $A = \{a, c, e\}$, $B = \{1, b, 3, d\}$, $C = \{\otimes, 2, \oplus\}$ jsou disjunktní po dvou.

Snadno zjistíme, že v případě dvou množin oba pojmy splývají a pro více než dvě množiny vyplývá z disjunktnosti po dvou disjunktnost. V obecném případě obrácené tvrzení neplatí. Následující diagramy demonstrují rozdíl mezi oběma pojmy.



$$A \cap B \cap C = \emptyset \\ (A \cap C \neq \emptyset, B \cap C \neq \emptyset)$$



$$A \cap B = \emptyset, A \cap C = \emptyset, B \cap C = \emptyset$$

Základní vlastnosti operací průnik a sjednocení lze formulovat následovně:

Komutativní zákony

$$A \cup B = B \cup A, \quad A \cap B = B \cap A.$$

Asociativní zákony

$$A \cup (B \cap C) = (A \cup B) \cap C, \quad A \cap (B \cup C) = (A \cap B) \cup C.$$

Distributivní zákony

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C), \quad A \cap (B \cup C) = (A \cap B) \cup (A \cap C).$$

Zákony idempotence

$$A \cup A = A, \quad A \cap A = A.$$

Zákony absorpce

$$A \cup (A \cap B) = A, \quad A \cap (A \cup B) = A.$$

Jak plyne z asociativních zákonů, můžeme určovat sjednocení či průnik většího počtu množin, aniž bychom při tom užívali závorek. V tomto případě používáme následující značení:

$$\bigcup_{i=1}^n A_i = A_1 \cup A_2 \cup \dots \cup A_n, \quad \bigcap_{i=1}^n A_i = A_1 \cap A_2 \cap \dots \cap A_n.$$

Definice (rozklad množiny)

Systém množin $\mathfrak{R} = \{A_1, \dots, A_n\}$ nazveme rozkladem množiny A , jestliže platí:

$$\bullet \quad \forall i, j \in \{1, \dots, n\} \quad [i \neq j \longrightarrow A_i \cap A_j = \emptyset],$$

tedy množiny A_1, \dots, A_n jsou disjunktní po dvou,

$$\bullet \quad A = \bigcup_{i=1}^n A_i,$$

(jejich sjednocení dává celou množinu A).

Množiny $A_i, i = 1, \dots, n$ nazýváme třídy rozkladu.

Příklad

- Systém množin $A_1 = \{1, 3, 5, 7\}, A_2 = \{2, 4, 6\}$ tvoří rozklad množiny $A = \{1, 2, 3, 4, 5, 6, 7\}$.
- Systém $A_0 = \{4m \mid m \in \mathbb{Z}\}, A_1 = \{4m + 1 \mid m \in \mathbb{Z}\}, A_2 = \{4m + 2 \mid m \in \mathbb{Z}\}, A_3 = \{4m + 3 \mid m \in \mathbb{Z}\}$ tvoří rozklad množiny celých čísel \mathbb{Z} .

Další operace s množinami, univerzum

Víme, že prvky množin mohou být zcela libovolné objekty. Uvažujeme-li však prakticky o množinách, zpravidla se omezujeme jen na množiny určitého typu. Počítáme-li s čísly, omezujeme se na množiny tvořené čísly. Bývá tedy vhodné brát všechny množiny, kterými se právě zabýváme, za podmnožiny jedné zvolené množiny. Při úvahách o číselných množinách to může podle potřeby být množina všech celých čísel, množina všech reálných čísel nebo množina všech komplexních čísel. Takovou množinu pak nazýváme univerzum a značíme U . Ještě jednou zdůrazněme, že nejde o nějaké absolutno, o nějakou „množinu všech množin“. Univerzum je to, co jsme si zvolili podle své potřeby.

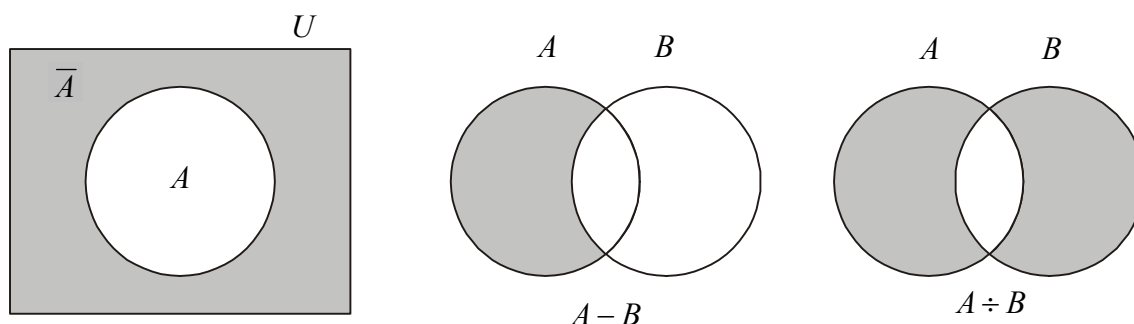
Definice (doplňěk množiny, rozdíl množin, symetrický rozdíl množin)

Nechť A, B jsou dvě libovolné podmnožiny univerza U .

- Doplňěk množiny A označujeme \bar{A} a definujeme jako množinu všech prvků univerza U , které nejsou obsaženy v množině A , tj. $\bar{A} = \{x \in U \mid x \notin A\}$.
- Rozdíl množin A, B označujeme $A - B$ (někdy také $A \setminus B$) a definujeme jako množinu všech prvků množiny A , které nejsou prvky B , tj. $A - B = \{x \in A \mid x \notin B\}$.
- Symetrický rozdíl množin A, B označujeme $A \div B$ (někdy také $A \Delta B$) a definujeme jako množinu prvků obsažených v právě jedné z množin A , nebo B , tj.

$$A \div B = (A - B) \cup (B - A).$$

Výše zavedené pojmy znázorňují následující Vennovy diagramy



a platí:

- $\overline{\bar{A}} = A$,
- $A - B = A \cap \bar{B}$,
- $\overline{A \cup B} = \bar{A} \cap \bar{B}$, $\overline{A \cap B} = \bar{A} \cup \bar{B}$. (de Morganova pravidla)

Poznámka (důkazy množinových identit)

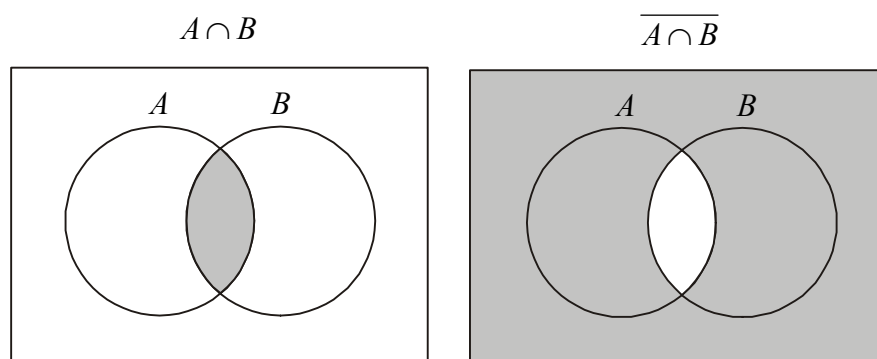
Důkazy množinových identit se běžně provádí některým z následujících postupů:

- Využitím Vennových diagramů.

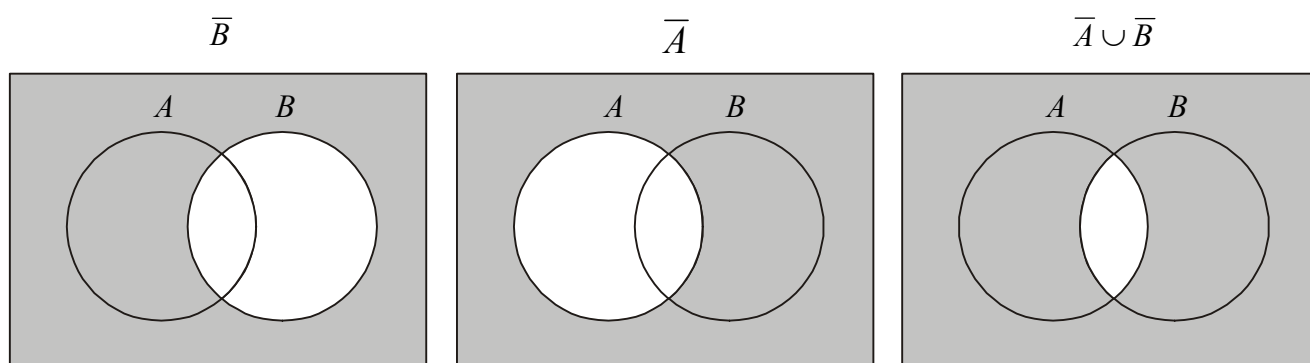
Pomocí Vennových diagramů znázorníme zvlášť levou a pravou část identity. Oba diagramy porovnáme.

Například důkaz de Morganova pravidla $\overline{A \cap B} = \overline{A} \cup \overline{B}$ může vypadat následovně.

Levá strana:



Pravá strana:



- Pomocí tabulky náležení.

Sestavíme tzv. tabulku náležení, která obsahuje sloupce reprezentující množiny, které se v dokazované identitě vyskytují a dále levou a pravou část identity. Řádky reprezentují všechny možné kombinace náležení do množin (1 – je prvkem, 0 – není prvkem). Pokud sloupce reprezentující levou a pravou část identity obsahují stejné hodnoty, identita platí.

Například důkaz de Morganova pravidla $\overline{A \cup B} = \overline{A} \cap \overline{B}$ může vypadat následovně:

| A | B | $A \cup B$ | $\overline{A \cup B}$ | \overline{A} | \overline{B} | $\overline{A} \cap \overline{B}$ |
|-----|-----|------------|-----------------------|----------------|----------------|----------------------------------|
| 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 |

- Pomocí definice rovnosti množin.

Označíme-li L množinu reprezentující levou stranu identity a P pravou stranu, potom stačí dokázat, že $L \subseteq P$ a současně $P \subseteq L$.

Definice (potenční množina)

Množina všech podmnožin dané množiny A je její potenční množina $P(A)$.

(Někdy také nazývaná boolean množiny A a označovaná $B(A)$, resp. 2^A .)

Platí:

- Je-li $|A| = n$ (tedy A je konečná množina), potom $|P(A)| = 2^n$.
- Pro libovolnou množinu A je $|A| < |P(A)|$.

První část uvedeného tvrzení dokážeme indukcí podle počtu prvků množiny A , tj. dle n .

Je-li $n = 0$, pak $A = \emptyset$ a $P(A) = \{\emptyset\}$, tedy množina, jejímž jediným prvkem je prázdná množina. Počet jejích prvků je $2^0 = 1$.

Nechť nyní $n \geq 1$ a předpokládejme, že potenční množina množiny o $n-1$ prvcích má 2^{n-1} prvků. Mějme množinu A o n prvcích a zvolme libovolně prvek $a \in A$. Potenční množina $P(A)$ je nyní sjednocení množiny G všech podmnožin množiny A , které neobsahují prvek a a množiny R takových podmnožin, které ho obsahují. Množina G je zřejmě potenční množinou množiny $A - \{a\}$ o $n-1$ prvcích a tedy (dle indukčního předpokladu) má 2^{n-1} prvků. Každou množinu $R \in R$ můžeme vyjádřit jako $R = G \cup \{a\}$, kde $G \in G$. Takovýchto množin je rovněž 2^{n-1} a tedy celkový počet prvků množiny $P(A)$ je $|P(A)| = 2^{n-1} + 2^{n-1} = 2^n$.

Příklady

- Pro $A = \emptyset$ dostáváme $P(A) = \{\emptyset\}$ a tedy $|P(A)| = 2^0 = 1$.
- Pro $A = \{\emptyset\}$ dostáváme $P(A) = \{\emptyset, \{\emptyset\}\}$ a tedy $|P(A)| = 2^1 = 2$.
- Pro $A = \{a, b\}$ dostáváme $P(A) = \{\emptyset, \{a\}, \{b\}, \{a, b\}\}$ a tedy $|P(A)| = 2^2 = 4$.

1.2 Relace

Kartézský součin množin

Pro definici pojmu relace, potřebujeme především pojem uspořádané n -tice prvků (n je dané přirozené číslo). Uspořádanou n -tici zapisujeme ve tvaru (a_1, \dots, a_n) , kde a_1, \dots, a_n jsou prvky nějaké množiny (prvky se mohou v uspořádané n -tici opakovat). Termín uspořádaná používáme proto, že záleží na pořadí prvků v n -tici. Například $(0,1,0,1)$, $(0,1,0,0)$ jsou dvě různé uspořádané čtveřice. Zdůrazněme, že uspořádané n -tice budeme vždy zapisovat do okrouhlých (terminologicky nesprávně kulatých) závorek.

Definice (kartézský součin)

Nechť A_1, \dots, A_n jsou množiny. Kartézský součin množin A_1, \dots, A_n označujeme $A_1 \times \dots \times A_n$ a definujeme jako množinu všech uspořádaných n -tic (a_1, \dots, a_n) , kde $a_i \in A_i, i = 1, \dots, n$. V případě, kdy $A_i = A, i = 1, \dots, n$ (tj. množiny A_1, \dots, A_n jsou stejné) mluvíme o n -té kartézské mocnině A a označujeme je A^n .

(Podobně jako kartézská soustava souřadnic má kartézský součin své jméno odvozeno od jména René Descartes (1596-1650), v polatinštlé podobě Cartesius.)

Lze tedy psát

$$A_1 \times \dots \times A_n = \{(a_1, \dots, a_n) \mid a_i \in A_i, i = 1, \dots, n\},$$
$$A^n = \underbrace{A \times \dots \times A}_{n\text{-krát}} = \{(a_1, \dots, a_n) \mid a_i \in A, i = 1, \dots, n\}.$$

Příklad

Pro množiny $A = \{a, b\}$, $B = \{1, 2, 3, 4\}$ dostáváme:

$$A \times B = \{(a,1), (a,2), (a,3), (a,4), (b,1), (b,2), (b,3), (b,4)\},$$

$$B \times A = \{(1,a), (2,a), (3,a), (4,a), (1,b), (2,b), (3,b), (4,b)\},$$

tedy $A \times B \neq B \times A$.

$$A^2 = \{(a,a), (a,b), (b,a), (b,b)\},$$

$$A^3 = \{(a,a,a), (a,a,b), (a,b,a), (b,a,a), (b,b,a), (a,b,b), (b,a,b), (b,b,b)\}.$$

Nyní již můžeme přistoupit k definici pojmu relace.

Definice (n -ární relace)

Nechť A_1, \dots, A_n jsou množiny. Libovolnou podmnožinu kartézského součinu $A_1 \times \dots \times A_n$ nazveme n -ární relací (nebo relací o aritě n) mezi množinami A_1, \dots, A_n .

Speciálně, podmnožiny n -té kartézské mocniny A^n nazýváme n -ární relace na množině A .

(Použité slovo arita je jen odtržená koncovka. Je to tak, že pro $n = 2, \dots, 10$ mají relace arity n označení podle násobných latinských číslovek: binární, ternární, kvaternární, kvinární, senární, septenární, oktonární, novenární, decenární.)

Poznamenejme, že v další části skript se budeme zabývat výhradně binárními relacemi!

Co to tedy ta relace je? Zavedli jsme něco zvláštního, co jsme nazvali relací, a zatím nevíme, k čemu to je. Přitom relace je prostě nějaký vztah a podmnožina kartézského součinu je jeho matematickou formalizací. Například, nechť A je množina nějakých hudebníků, řekněme

$$A = \{\text{Jan, Josef, Pavel, Petr, Václav}\},$$

B množina nějakých hudebních nástrojů, třeba

$$B = \{\text{klavír, housle, kontrabas, fagot}\}.$$

Nyní můžeme vzít relaci „hudebník a umí hrát na nástroj b “ a označit ji třeba H . Jde samozřejmě o relaci mezi prvky množiny A a prvky množiny B (relace z množiny A do B), tedy o podmnožinu kartézského součinu $A \times B$, tj. o množinu uspořádaných dvojic (a, b) takových, že $a \in A, b \in B$ (hudebník a umí hrát na b). Může to vypadat takto:

$$H = \{(\text{Jan, housle}), (\text{Jan, fagot}), (\text{Josef, klavír}), (\text{Josef, housle}), (\text{Pavel, kontrabas}), \\ (\text{Pavel, fagot}), (\text{Petr, klavír}), (\text{Václav, housle}), (\text{Václav, kontrabas})\}.$$

Je to relace arity 2, tedy binární.

Poznámka (zadávání (binárních) relací)

Relaci lze zadat některým z následujících způsobů:

- Výčtem všech jejích prvků.
- Maticí sousednosti.

Je-li R relace na n -prvkové množině $A = \{a_1, \dots, a_n\}$, potom matice sousednosti

$M_R = (m_{i,j})_{i,j=1}^n$ relace R je definována následovně:

$$m_{i,j} = \begin{cases} 1, & \text{jestliže } (a_i, a_j) \in R, \\ 0, & \text{jinde.} \end{cases}$$

- Orientovaným grafem (viz úvod kapitoly 3. Teorie grafů).

Příklad (relace dělitelnosti)

Důležitá relace na množině kladných přirozených čísel N^+ je dělitelnost, tedy „číslo a je dělitelem čísla b “ (označujeme $a|b$). Pro jednoduchost se omezíme na množinu přirozených čísel $\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$.

Tato relace je určena následujícím výčtem všech jejích prvků (tj. dvojic (a, b) , kde a dělí beze zbytku b):

$$\{(1,1), (1,2), (1,3), (1,4), (1,5), (1,6), (1,7), (1,8), (1,9), (1,10), (2,2), (2,4), (2,6), (2,8), (2,10), (3,3), (3,6), (3,9), (4,4), (4,8), (5,5), (5,10), (6,6), (7,7), (8,8), (9,9), (10,10)\}.$$

Její matice sousednosti $M_{|}$ vypadá takto:

$$M_{|} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Poznámka

- Relace jsou množiny a proto je budeme často označovat velkými písmeny stejně jako množiny (R , S , T). Nemusí tomu tak být vždy. Například se užívá i malých řeckých písmen ρ , τ , někdy také různých speciálních symbolů jako \equiv , \cong , \prec , \triangleleft .
- Máme-li relaci R , pak skutečnost, že dvojice (a, b) patří do relace R zapisujeme $(a, b) \in R$ nebo také aRb .

- Speciálními typy relací jsou:
 - Prázdná relace na A , tj. prázdná podmnožina kartézského součinu $A \times A$.
 - Úplná (univerzální) relace na množině A , která je rovna A^2 .
 - Identická relace $I_A = \{(a, a) \mid a \in A\}$, někdy též nazývaná diagonální relace.

Vlastnosti relací

Předpokládejme, že ρ je relace na množině A (tj. $\rho \subseteq A \times A$) a definujme si některé důležitější vlastnosti relací.

Definice (reflexivita, symetrie, antisymetrie, tranzitivita)

Nechť ρ je binární relace na množině A . Potom:

- Řekneme, že relace ρ je reflexivní, jestliže každý prvek množiny A je v relaci sám se sebou. Uvedenou vlastnost lze vyjádřit zápisem

$$\forall a [a \in A \rightarrow (a, a) \in \rho].$$

(Podrobněji k uvedenému způsobu zápisu viz odstavec 2.3 Základy predikátové logiky.)

- Řekneme, že binární relace ρ je symetrická, jestliže s každou uspořádanou dvojicí (a, b) prvků množiny A , která je v relaci ρ , je v dané relaci i dvojice (b, a) .

Uvedenou vlastnost vyjádříme zápisem

$$\forall a, b \in A [(a, b) \in \rho \rightarrow (b, a) \in \rho].$$

- Řekneme, že binární relace ρ je antisymetrická, jestliže platí: je-li v relaci dvojice (a, b) a současně (b, a) , potom $a = b$.

Uvedenou vlastnost vyjádříme zápisem

$$\forall a, b \in A [(a, b) \in \rho \wedge (b, a) \in \rho \rightarrow a = b].$$

- Řekneme, že binární relace ρ je tranzitivní, jestliže platí: je-li v relaci dvojice (a, b) a současně (b, c) , potom je v relaci i dvojice (a, c) .

Uvedenou vlastnost vyjádříme zápisem

$$\forall a, b, c \in A [(a, b) \in \rho \wedge (b, c) \in \rho \rightarrow (a, c) \in \rho].$$

Základní představu o vlastnostech nejdůležitějších relací si lze udělat z dále uvedené tabulky:

| množina | relace | reflexivita | symetrie | antisymetrie | tranzitivita |
|------------------------------|-----------------|-------------|----------|--------------|--------------|
| libovolná $A \neq \emptyset$ | $=$ | ano | ano | ano | ano |
| libovolná $A \neq \emptyset$ | \neq | ne | ano | ne | ne |
| reálná čísla | \leq | ano | ne | ano | ano |
| reálná čísla | $<$ | ne | ne | ano | ano |
| kladná přirozené | „býti dělitel“ | ano | ne | ano | ano |
| nenulová celá | „býti dělitel“ | ano | ne | ne | ano |
| celá čísla | $\text{mod } m$ | ano | ano | ne | ano |
| množinový systém | \subseteq | ano | ne | ano | ano |
| množinový systém | \subset | ne | ne | ano | ano |

Zavedené pojmy (reflexivita, symetrie apod.) se využívají při definici základních typů relací, z nichž nejdůležitější je relace ekvivalence a relace částečného uspořádání.

Definice (relace ekvivalence)

Relace na množině A , která je reflexivní, symetrická a tranzitivní, se nazývá relace ekvivalence na množině A .

S relací ekvivalence úzce souvisí již dříve zavedený pojem rozklad množiny. Platí totiž, že každá relace ekvivalence na A definuje rozklad A a obráceně. Každý rozklad množiny A definuje relaci ekvivalence na A .

Je-li totiž ρ ekvivalence na A , potom systém všech různých množin $[a] = \{b \in A \mid (a, b) \in \rho\}$ (tzv. třídy ekvivalence, tj. množiny všech navzájem ekvivalentních prvků A) tvoří zřejmě rozklad A (zdůvodněte!). Naopak, je-li A_1, \dots, A_n rozklad A , potom definujeme relaci ekvivalence ρ následovně $(a, b) \in \rho$ právě tehdy, jestliže existuje třída rozkladu A_i taková, že současně a i b jsou prvky A_i (tj. $a, b \in A_i$).

Poznámka (kongruence modulo m)

Velmi důležitým příkladem relace ekvivalence je tzv. relace kongruence modulo m (m přirozené, $m \geq 2$). Tato relace je definována na množině všech celých čísel Z následovně:

Řekneme, že celá čísla a, b jsou kongruentní modulo m , píšeme $a \equiv b \pmod{m}$, resp. $a \equiv_m b$, jestliže obě čísla dávají při dělení číslem m stejný zbytek (ekvivalentně, rozdíl $a - b$ je celočíselným násobkem m). Snadno ověříme, že uvedená relace je skutečně reflexivní, symetrická, tranzitivní a třídy ekvivalence (tvořící rozklad množiny Z) jsou

$$[0] = \{m \cdot t \mid t \in Z\},$$

$$[1] = \{m \cdot t + 1 \mid t \in Z\}$$

\vdots

$$[m-1] = \{m \cdot t + (m-1) \mid t \in Z\}.$$

Například pro $m = 2$ dostáváme dvě třídy ekvivalence

$$[0] = \{2t \mid t \in Z\}, \quad [1] = \{2t + 1 \mid t \in Z\},$$

které odpovídají rozkladu celých čísel na sudá a lichá.

Definice (relace částečného uspořádání)

Řekneme, že relace \leq definovaná na množině A je relace částečného uspořádání na množině A (někdy se říká pouze uspořádání), jestliže \leq je reflexivní, antisymetrická a tranzitivní. Dvojici (A, \leq) nazýváme částečně uspořádaná množina, neboli poset (zkratka partially ordered set).

Pozor. Symbol \leq budeme běžně používat ve dvou významech. Jednak jako označení pro obecnou relaci uspořádání, ale i jako označení tzv. přirozeného uspořádání číselných množin. Skutečný význam bude vždy zřejmý z kontextu.

Nejjednodušším příkladem uspořádání je již zmíněné přirozené uspořádání \leq množiny přirozených (ale i celých, racionálních, reálných, nikoliv však komplexních) čísel. Dalšími příklady uspořádání jsou již zmíněná relace dělitelnosti označovaná symbolem \mid ($a \mid b$ čteme „ a dělí b “) a definovaná na množině všech kladných přirozených čísel $N^+ = \{1, 2, \dots\}$, resp. relace býti podmnožinou \subseteq definovaná na nějakém systému množin (např. potenční množině). Ověřte, že v uvedených případech jde skutečně o uspořádání a zdůvodněte, proč relace dělitelnosti \mid není uspořádání na množině celých čísel.

Poznámka

Relaci uspořádání lze skutečně interpretovat jako jisté zobecněné porovnávání, které nám umožňuje uspořádat prvky dané množiny podle „velikosti“. Důležité je si však uvědomit, že nemusí jít o porovnávání na které jsme běžně zvyklí, například z oboru reálných čísel, kde je

možné pro libovolná dvě čísla rozhodnout, které z nich je větší, resp. zda jsou sobě rovna. Například v rámci dělitelnosti | nejsou čísla 3 a 7 „porovnatelná“ a neboť 3 nedělí beze zbytku 7 ani 7 nedělí beze zbytku 3. Stejně tak při uspořádání množin inkluzí \subseteq nemusí být dané dvě množiny v relaci a tedy porovnatelné.

Definice (předchůdce, následník, minimální, nejmenší, maximální, největší prvek)

Nechť (A, \leq) je poset. Potom řekneme, že:

- Prvek $a \in A$ je předchůdce prvku $b \in A$ (resp. b je následník prvku a) jestliže platí:
 - $a \leq b$,
 - je-li $a \leq c \leq b$, potom $c = a$ nebo $c = b$.
- Prvek $a \in A$ je minimální prvek A , jestliže pro každé $b \in A$ platí: je-li $b \leq a$, potom $b = a$.
- Prvek $a \in A$ je nejmenší prvek A , jestliže pro každé $b \in A$ platí $a \leq b$. (Pokud nejmenší prvek existuje, je určen jednoznačně.)
- Prvek $a \in A$ je maximální prvek A , jestliže pro každé $b \in A$ platí: je-li $a \leq b$, potom $b = a$.
- Prvek $a \in A$ je největší prvek A , jestliže pro každé $b \in A$ platí $b \leq a$. (Pokud největší prvek existuje, je určen jednoznačně.)

Ke grafickému znázornění uspořádaných množin se užívá tzv. Hasseův diagram. Jde o náčrt grafu (podrobnosti k pojmu graf viz kapitola 3. Teorie grafů), jehož množinu uzlů tvoří prvky množiny A a hrany spojují pouze předchůdce a následníky. Při kreslení je třeba dodržet pravidlo: je-li a předchůdce b , potom a nakreslíme níže než b .

Příklad

Nakreslete Hasseův diagram následujících uspořádaných množin a určete minimální a maximální prvky, největší a nejmenší prvek.

- Uspořádaná množina (A, \leq) , kde $A = \{1, 2, 3, 4, 5, 6\}$ a „ \leq “ je přirozené uspořádání.

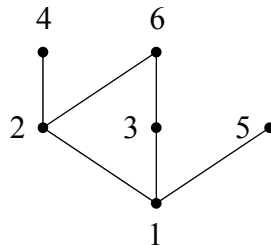


Z Hasseova diagramu snadno zjistíme, že:

1 je nejmenší prvek a tedy i jediný minimální prvek,

6 je největší a tedy i jediný maximální prvek.

- Uspořádaná množina $(A, |)$, kde $A = \{1, 2, 3, 4, 5, 6\}$ a $|$ je relace „býti dělitelem“.

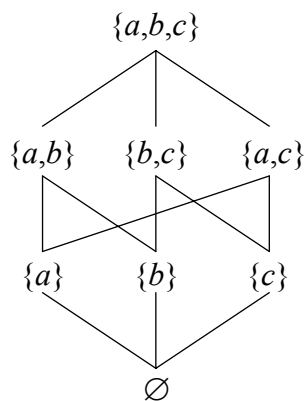


Z Hasseova diagramu snadno zjistíme, že:

1 je nejmenší prvek a tedy i jediný minimální prvek,

4, 5, 6 jsou maximální prvky a tedy neexistuje největší prvek.

- Uspořádaná množina $(P(A), \subseteq)$, kde $P(A)$ je potenční množina množiny $A = \{a, b, c\}$ a \subseteq je relace býti podmnožinou.



Z Hasseova diagramu snadno zjistíme, že:

\emptyset je nejmenší prvek a tedy i jediný minimální prvek,

$\{a, b, c\}$ je největší a tedy i jediný maximální prvek.

1.3 Zobrazení

Jedním z nejdůležitějších pojmů matematiky je zobrazení. V podstatě jde o speciální relaci.

Definice (zobrazení)

Nechť A, B jsou množiny a $f \subseteq A \times B$ relace na A, B . Řekneme, že f je zobrazení z množiny A do B , píšeme $f : A \rightarrow B$, jestliže platí: je-li $(a, b) \in f$ a současně $(a, c) \in f$, potom $b = c$.

(Tj. ke každému $a \in A$ existuje nejvýše jedno $b \in B$ takové, že $(a, b) \in f$.)

Pro zobrazení používáme místo zápisu $(a, b) \in f$ zápis $b = f(a)$ a říkáme, že b je hodnota zobrazení f v bodě a , resp. že b je obraz prvku a v zobrazení f . Prvku a také říkáme vzor prvku b .

Poznámka

- Místo „zobrazení z množiny A “ můžeme někdy říci „zobrazení množiny A “ (bez předložky), a to tehdy, jestliže každý prvek množiny A má v tomto zobrazení svůj obraz.
- Je-li $f : A \rightarrow B$ zobrazení a $S \subseteq A$, potom množinu $f(S) = \{b \in B \mid \exists a \in A f(a) = b\}$ nazýváme obraz množiny S při zobrazení f .
- Zobrazení do množiny čísel se zpravidla nazývá funkcionál, zobrazení z množiny čísel opět do množiny čísel se nazývá funkce.

Například funkce $y = 2x + 3$ je zobrazení množiny reálných čísel R opět na množinu R , funkce $y = x^2$ je zobrazení množiny R do množiny R (přesněji zobrazení množiny R na množinu všech nezáporných reálných čísel R^+). Funkce $y = \ln x$ (tzv. přirozený logaritmus) je zobrazení z množiny R na množinu R (přesněji zobrazení množiny R^+ na množinu R).

Definice (prosté, „na“, vzájemně jednoznačné a inverzní zobrazení)

- Řekneme, že zobrazení $f : A \rightarrow B$ je prosté (injektivní) zobrazení A do B , jestliže

$$f(a_1) = f(a_2) \rightarrow a_1 = a_2,$$

tj. různé vzory mají různé obrazy.

- Řekneme, že zobrazení $f : A \rightarrow B$ je zobrazení na množinu B (surjektivní, „na“), jestliže

$$\forall b \in B \exists a \in A \quad f(a) = b,$$

tj. každý prvek z B je obrazem nějakého prvku z A .

- Řekneme, že zobrazení $f : A \rightarrow B$ je vzájemně jednoznačné zobrazení A na B (bijektivní, „1-1“), jestliže je prosté a „na“.
- Nechť $f : A \rightarrow B$ je prosté zobrazení. Potom zobrazení $f^{-1} : B \rightarrow A$ definované vztahem

$$a = f^{-1}(b) \longleftrightarrow b = f(a)$$

nazveme inverzní zobrazení k zobrazení f .

(POZOR! Na rozdíl od relací, kde inverzní relace existuje vždy, existuje inverzní zobrazení pouze k prostému zobrazení.)

Zavedeme-li ekvivalenci množin (množiny A, B nazveme ekvivalentní, jestliže existuje prosté zobrazení množiny A na množinu B), můžeme již zcela exaktně zavést pojem konečné a nekonečné množiny. Říkáme, že množina A je nekonečná, je-li ekvivalentní s některou svou vlastní podmnožinou. Ekvivalence mezi nekonečnými množinami je pak základem pro zavedení nekonečných kardinálních čísel (např. již dříve zmíněné symboly \aleph_0, \aleph_1). V opačném případě (tj. kdy A není ekvivalentní s žádnou svou vlastní podmnožinou) říkáme, že A je konečná množina. Snadno tak zjistíme, že množina všech celých čísel \mathbb{Z} je nekonečná, protože funkce $y = 2n$ je jejím prostým zobrazením na množinu všech sudých celých čísel, což je samozřejmě její vlastní podmnožina. Stejně snadno si uvědomíme, že dvě konečné množiny jsou ekvivalentní právě tehdy, mají-li tentýž počet prvků.

Nechť nyní A, B jsou konečné množiny, potom počet zobrazení množiny A (bez předložky!) do množiny B je roven $|B|^{|A|}$. Je-li $A \neq \emptyset$, je tvrzení jasné. Pro každý prvek množiny A máme $|B|$ možností, jak mu přiřadit obraz, tedy celkem $|B|^{|A|}$ možností, jak zobrazení určit. Do rozpaků nás může uvést případ, kdy $A = \emptyset$. Skutečně však můžeme (formálně) tvrdit, že existuje právě jedno zobrazení prázdné množiny do libovolné množiny B (včetně $B \neq \emptyset$). Měla by to být množina uspořádaných dvojic (a, b) , kde a je prvek prázdné množiny a b je jeho obraz v množině B . Prázdné zobrazení, tj. zobrazení, které je prázdnou množinou, toto splňuje. Žádný prvek prázdné množiny neexistuje, tudíž neexistuje ani žádná dvojice (a, b) , tedy prázdné zobrazení je vskutku požadovanou množinou dvojic. To odpovídá vztahu

$n^0 = 1$, pro každé celé $n \geq 0$, tedy i $0^0 = 1$. Něco jiného je ovšem 0^n pro $n \geq 1$. Je-li $A \neq \emptyset, B = \emptyset$, pak zobrazení množiny A do množiny B by mělo obsahovat všechny dvojice (a, b) , kde $a \in A$ a b je jeho obraz v prázdné množině \emptyset . Žádný obraz neexistuje, tedy neexistuje ani požadované zobrazení a skutečně $|B|^{|A|} = 0^{|A|} = 0$. (Vidíme tady rozdíl mezi případem, kdy zobrazení neexistuje a kdy existuje, ale je prázdné. Vcelku se dá říci, že o prvcích prázdné množiny můžeme tvrdit cokoliv, kromě toho, že existují.) Z výše uvedeného důvodu se běžně množina všech zobrazení množiny A do množiny B (i pro nekonečné množiny) značí B^A .

Poznámka (funkce důležité v oblasti diskrétní matematiky)

- Funkci $f: Z \rightarrow N$ definovanou vztahem

$$f(n) = n \bmod m,$$

kde $m \geq 2$ je přirozené číslo a $n \bmod m$ označuje zbytek při dělení čísla n číslem m nazveme mod funkcí.

- Funkci $i_A: A \rightarrow A$ definovanou na libovolné množině A vztahem

$$i_A(a) = a, \text{ pro libovolné } a \in A,$$

nazveme identickou (jednotkovou) funkcí na A .

- Funkci $\lfloor \cdot \rfloor: R \rightarrow Z$ definovanou na množině reálných čísel jako největší celé číslo menší nebo rovné x nazveme dolní celou částí x . Tuto funkci lze definovat vlastnostmi:

- $\forall x \in R \quad \lfloor x \rfloor \in Z,$
- $\forall x \in R \quad \lfloor x \rfloor \leq x < \lfloor x \rfloor + 1.$

- Funkci $\lceil \cdot \rceil: R \rightarrow Z$ definovanou na množině reálných čísel jako nejmenší celé číslo větší nebo rovné x nazveme horní celou částí x . Tuto funkci lze definovat vlastnostmi:

- $\forall x \in R \quad \lceil x \rceil \in Z,$
- $\forall x \in R \quad \lceil x \rceil - 1 < x \leq \lceil x \rceil.$

- Funkci $\{ \cdot \}: R \rightarrow R$ definovanou vztahem $\{x\} = x - \lfloor x \rfloor$ nazveme lomenou částí x .

Například platí:

$$\begin{array}{llll} 3 = 36 \bmod 11, & 7 = -45 \bmod 13, & \lfloor 3,72 \rfloor = 3, & \lfloor -3,72 \rfloor = -4, \\ \lceil 6,12 \rceil = 7, & \lceil -6,12 \rceil = -6, & \{5,27\} = 0,27, & \{-5,27\} = 0,73. \end{array}$$

1.4 Operace

Pojem operace jsme si už intuitivně vyložili. Formálně lze operaci definovat jako jisté zobrazení následovně.

Definice

n -ární operaci na množině A rozumíme každé zobrazení $\rho : A^n \rightarrow A$, kde $n \in \mathbb{N}$ a A^n je n -tá kartézská mocnina množiny A . n -ární operace přiřazuje uspořádaným n -ticím $(a_1, \dots, a_n) \in A^n$ prvek $\rho(a_1, \dots, a_n) \in A$ označovaný jako výsledek operace.

(Velmi často budeme operace značit speciálními symboly, např. $*$, $+$, $-$, \ddot{a} , μ , \odot , $\dot{\cup}$, \bullet , \otimes , \oplus)

Poznámka (binární, unární a nulární operace)

Operace mohou být libovolné arity (nalezení těžiště trojúhelníku s danými vrcholy A, B, C je příkladem operace arity 3, tedy ternární operace). Nejčastěji se však setkáme s případy:

- $n = 2$, kdy mluvíme o binární operaci (zobrazení $\rho : A^2 \rightarrow A$),
- $n = 1$, kdy mluvíme o unární operaci (zobrazení $\rho : A \rightarrow A$),
- $n = 0$, kdy mluvíme o nulární operaci (což je vybraný význačný prvek nějaké množiny, např. jednotkový prvek grupy).

V další části se stručně zmíníme o základních vlastnostech binárních operací.

Definice (asociativita, komutativita, neutrální a symetrický prvek)

Řekneme, že binární operace $*$: $A^2 \rightarrow A$ je:

- asociativní, jestliže

$$\forall a_1, a_2, a_3 \in A \text{ platí } (a_1 * a_2) * a_3 = a_1 * (a_2 * a_3),$$

- komutativní, jestliže

$$\forall a_1, a_2 \in A \text{ platí } a_1 * a_2 = a_2 * a_1.$$

Dále řekneme, že prvek $e \in A$ je jednotkový prvek operace $*$, jestliže

$$\forall a \in A \text{ platí } a * e = e * a = a.$$

Řekneme dále, že k prvku $a \in A$ existuje symetrický prvek $a' \in A$, jestliže

$$a * a' = a' * a = e,$$

kde e je neutrální prvek .

(Velmi často se místo o symetrickém prvku mluví o inverzním nebo opačném prvku. Inverzní prvek označujeme a^{-1} a opačný $-a$.)

Příklad

- Nejznámějšími příklady binárních operací jsou obyčejné sčítání a násobení reálných čísel. Jejich vlastnosti jsou:

„+“ ... asociativní, komutativní, existuje neutrální prvek 0 a opačný prvek $-a$,

„×“ ... asociativní, komutativní, existuje neutrální prvek 1 a ke každému nenulovému číslu existuje inverzní prvek a^{-1} .

- Dalším velmi důležitým příkladem operací jsou operace sčítání \oplus a násobení \otimes modulo m , definované na množině celých čísel Z následovně:

$$a \oplus b = (a + b) \bmod m,$$

$$a \otimes b = (a \times b) \bmod m.$$

Jejich vlastnosti jsou:

„ \oplus “ ... asociativní, komutativní, existuje neutrální prvek 0 a opačný prvek $-a \bmod m$,

„ \otimes “ ... asociativní, komutativní, existuje neutrální prvek 1, ovšem v případě obecného modulu m neexistuje inverzní prvek (podrobněji viz druhý díl těchto skript – Diskrétní matematika II). Tuto skutečnost dokládá následující ukázka.

Jako ukázku uvádíme tabulku sčítání a násobení modulo 6.

| \oplus | 0 | 1 | 2 | 3 | 4 | 5 |
|----------|---|---|---|---|---|---|
| 0 | 0 | 1 | 2 | 3 | 4 | 5 |
| 1 | 1 | 2 | 3 | 4 | 5 | 0 |
| 2 | 2 | 3 | 4 | 5 | 0 | 1 |
| 3 | 3 | 4 | 5 | 0 | 1 | 2 |
| 4 | 4 | 5 | 0 | 1 | 2 | 3 |
| 5 | 5 | 0 | 1 | 2 | 3 | 4 |

| \otimes | 0 | 1 | 2 | 3 | 4 | 5 |
|-----------|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 2 | 3 | 4 | 5 |
| 2 | 0 | 2 | 4 | 0 | 2 | 4 |
| 3 | 0 | 3 | 0 | 3 | 0 | 3 |
| 4 | 0 | 4 | 2 | 0 | 4 | 2 |
| 5 | 0 | 5 | 4 | 3 | 2 | 1 |

Všimněme si, že k číslům 2, 3, 4 neexistují inverzní prvky, navíc $2 \otimes 3 = 3 \otimes 4 = 0$.

Kapitola 2. Základy matematické logiky

Vedle teorie množin patří k základům matematiky rovněž matematická logika, neboť matematické myšlení se musí řídit logikou. Logika je nauka o zákonitostech správného myšlení a zabýval se jí už Aristotelés (384 – 322 před Kr.). Matematicky ji zformalizoval George Boole (1815 - 1864).

2.1 Základy výrokové logiky

Základním pojmem výrokové logiky je výrok. Pro naši potřebu vystačíme s definicí, že výrok je tvrzení, o kterém lze jednoznačně rozhodnout, zda je či není pravdivé. Nejjednodušší výroky nazýváme atomické výroky a budeme je značit malými písmeny latinské abecedy, nejčastěji p, q, r, s . Jejich pravdivostní hodnotu (tj. zda je pravdivý, resp. nepravdivý) budeme značit 1 (pravda) nebo 0 (nepravda). Je-li tedy výrok p pravdivý, píšeme $p = 1$, je-li nepravdivý $p = 0$. Je to trochu nepřesné, neboť výrok ztotožňujeme s jeho pravdivostní hodnotou, což bychom mohli interpretovat, že všechny pravdivé výroky jsou si rovny a všechny nepravdivé rovněž. Přesnější by bylo mluvit o pravdivostním ohodnocení výroků, tedy o zobrazení množiny výroků do množiny $\{0, 1\}$. My se ale budeme držet běžně užívaného postupu.

Příklad

Následující tvrzení jsou výroky:

- Praha je hlavní město České republiky. (Pravdivý výrok.)
- $2 + 7 = 33$. (Nepravdivý výrok.)
- Dnes je 6. července 1415. (Nepravdivý výrok.)

Následující věty nejsou výroky (nelze určit jejich pravdivostní hodnotu):

- Kolik je hodin?
- Tato věta je nepravdivá.

(Libovolné pravdivostní ohodnocení vede ke sporu.)

- $x - 53 = 7$.

(Pozor, nezaměňovat s existencí řešení dané rovnice.)

Atomické výroky lze považovat za jakési stavební kameny výrokové logiky, podobné nejjednodušším větám přirozeného jazyka a stejně tak jako věty spojujeme do souvětí pomocí spojek, vytváříme pomocí atomických výroků a logických spojek (logických operátorů) složené výroky (jejich přesná definice bude teprve následovat). Mezi základní logické spojky patří především negace, konjunkce, disjunkce, implikace a ekvivalence. Jejich význam je následující.

Negace

Nechť p je výrok, potom symbolem $\neg p$, resp. \bar{p} , označujeme negaci výroku p , tedy výrok „není pravda, že platí p “. Pravdivostní hodnota negace je dána následující tzv. pravdivostní tabulkou.

| p | $\neg p$ |
|-----|----------|
| 0 | 1 |
| 1 | 0 |

Výrok $\neg p$, resp. \bar{p} čteme „non p “, resp. „negace p “.

Konjunkce

Nechť p , q jsou výroky, potom symbolem $p \wedge q$ označujeme konjunkci výroků p , q , tedy výrok „platí současně p i q “. V přirozeném jazyce tak logické spojce \wedge odpovídá spojka „a“. Pravdivostní hodnota konjunkce je dána následující pravdivostní tabulkou.

| p | q | $p \wedge q$ |
|-----|-----|--------------|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Výrok $p \wedge q$ čteme „ p et q “, resp. „ p konjunkce q “.

Disjunkce

Nechť p , q jsou výroky, potom symbolem $p \vee q$ označujeme disjunkci výroků p , q , tedy výrok „platí p nebo q “. V přirozeném jazyce tak logické spojce \vee odpovídá spojka „nebo“. Pravdivostní hodnota disjunkce je dána následující pravdivostní tabulkou.

| p | q | $p \vee q$ |
|-----|-----|------------|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

Výrok $p \vee q$ čteme některým z následujících způsobů: „ p nebo q “, „ p disjunkce q “, „ p vel q “. Dále poznamenejme, že tvrzení „platí p nebo q “ nevylučuje současnou platnost obou výroků, což vyjadřuje právě latinská spojka „vel“, kdežto latinská spojka „aut“ znamená, že nastává právě jeden z obou případů.

Implikace

Nechť p , q jsou výroky, potom symbolem $p \rightarrow q$ označujeme tzv. implikaci, tedy výrok „z p plyne q “, resp. „jestliže p , potom i q “. Pravdivostní hodnota implikace je dána následující pravdivostní tabulkou.

| p | q | $p \rightarrow q$ |
|-----|-----|-------------------|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Výrok $p \rightarrow q$ obvykle čteme jako „ p implikace q “, kde p označujeme jako premisu, resp. hypotézu a q jako konkluzi, resp. závěr.

Ekvivalence

Nechť p , q jsou výroky, potom symbolem $p \leftrightarrow q$ označujeme ekvivalenci výroků p , q , tedy výrok „ p právě když q “. Pravdivostní hodnota ekvivalence je dána následující pravdivostní tabulkou.

| p | q | $p \leftrightarrow q$ |
|-----|-----|-----------------------|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Výrok $p \leftrightarrow q$ obvykle čteme jako „ p ekvivalence q “.

Nyní lze přistoupit k exaktní definici pojmu složený výrok.

Definice (složený výrok)

Složeným výrokem nazveme každý výraz, který vznikne konečným počtem užití následujících pravidel:

1. Každý atomický výrok je složený výrok.
2. Jestliže p, q jsou složené výroky, potom

$$\left. \begin{array}{l} \neg p \\ p \wedge q \\ p \vee q \\ p \rightarrow q \\ p \leftrightarrow q \end{array} \right\} \text{ je složený výrok.}$$

Při vytváření složených výroků je možné využívat různé druhy závorek, které zvyšují přehlednost zápisu a umožňují také změnit pořadí (prioritu) vyhodnocování jednotlivých logických spojek. Pokud nejsou použity závorky, je priorita vyhodnocování logických spojek dána následující tabulkou:

| Logická spojka | Priorita |
|----------------------|----------|
| negace | 4 |
| konjunkce, disjunkce | 3 |
| implikace | 2 |
| ekvivalence | 1 |

Výrazy se vyhodnocují od nejvyšší priority k nejnižší a při stejné prioritě se postupuje zleva doprava.

Poznámka (pravdivostní tabulka složeného výroku)

Definice logických spojek a pravidla pro vytváření složených výroků jsou postačující k tomu, abychom byli schopni určit pravdivostní hodnotu libovolného složeného výroku (samozřejmě v závislosti na pravdivostním ohodnocení atomických výroků v něm vystupujících). K tomu se běžně užívají tzv. pravdivostní tabulky, které obvykle obsahuje sloupce reprezentující jednotlivé atomické výroky a sloupec obsahující pravdivostní hodnoty (ty doplňujeme) uvažovaného složeného výroku. Řádky tabulky pak reprezentují všechny možné kombinace pravdivostních hodnot atomických výroků a jim odpovídající pravdivostní hodnotu

uvažovaného složeného výroku (pravdivostní tabulka výroku s n atomickými výroky tak obsahuje 2^n řádků – zdůvodněte!).

Příklad

K následujícím výrokům sestrojte pravdivostní tabulky.

a) $p \vee \neg q \rightarrow q$

| p | q | $\neg q$ | $p \vee \neg q$ | $p \vee \neg q \rightarrow q$ |
|-----|-----|----------|-----------------|-------------------------------|
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |

b) $p \rightarrow q \wedge \neg p \rightarrow r$

Pozor, při vyhodnocování zadaného výroku je třeba zvážit priority logických spojek!

| p | q | r | $q \wedge \neg p$ | $p \rightarrow q \wedge \neg p$ | $p \rightarrow q \wedge \neg p \rightarrow r$ |
|-----|-----|-----|-------------------|---------------------------------|---|
| 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 | 0 | 1 |

V této souvislosti uveďme, že ke zvýšení přehlednosti lze použít i tzv. úplně uzávorkované výroky (závorky znázorňují pořadí vyhodnocování). Jde o výrazy (složené výroky), které jsou vytvořeny konečným počtem aplikací následujících pravidel:

1. Každý atomický výrok je úplně uzávorkovaný výrok.
2. Jestliže p, q jsou úplně uzávorkované výroky, potom také

$$\left. \begin{array}{l} (\neg p) \\ (p \wedge q) \\ (p \vee q) \\ (p \rightarrow q) \\ (p \leftrightarrow q) \end{array} \right\} \text{ jsou úplně uzávorkované výroky.}$$

Příklad

Následující neuzávorkované výroky převed'te na úplně uzavorkované. Uzávorkování proved'te tak, aby oba výroky byly vyhodnocovány stejným způsobem.

a) $p \wedge q \vee r \wedge s$

Všechny použité logické spojky mají stejnou prioritu, tedy daný výrok vyhodnocujeme zleva doprava. Dostáváme: $((p \wedge q) \vee r) \wedge s$

b) $\neg p \wedge q \vee \neg r$

Negace má nejvyšší prioritu, tedy: $((\neg p) \wedge q) \vee (\neg r)$

c) $p \leftrightarrow q \rightarrow r$

Implikace má vyšší prioritu než ekvivalence, tedy: $(p \leftrightarrow (q \rightarrow r))$

d) $p \rightarrow q \vee \neg p \rightarrow r$

Nejprve vyhodnotíme negaci, následně disjunkci a na závěr vyhodnotíme zleva implikace.

Dostáváme: $((p \rightarrow (q \vee (\neg p))) \rightarrow r)$

Poznámka (infixní a prefixní zápis výroků)

Doposud používaný způsob zápisu výroků se označuje jako infixní (logická spojka je umístěna mezi operandy) a jelikož jsme na něho zvyklí, považujeme ho proto za přirozený. Vyhodnocování jeho pravdivosti je však poměrně komplikované, neboť je nutné uvažovat priority jednotlivých logických spojek a její případnou změnu způsobenou uzavorkováním. Z tohoto pohledu je velmi výhodný tzv. (polský) prefixní tvar výroků (nevyužívá závorky!), který je definován následovně:

1. Každý atomický výrok je výrok v prefixním tvaru.

2. Jestliže p, q jsou výroky v prefixním tvaru, potom

$$\left. \begin{array}{l} \neg p \quad \text{je zápis negace výroku } p \\ \wedge pq \quad \text{je zápis konjunkce výroků } p, q \\ \vee pq \quad \text{je zápis disjunkce výroků } p, q \\ \rightarrow pq \quad \text{je zápis implikace výroků } p, q \\ \leftrightarrow pq \quad \text{je zápis ekvivalence výroků } p, q \end{array} \right\} \text{ v polském prefixním tvaru.}$$

Poznamenejme, že vyhodnocování pravdivosti výroků v prefixním tvaru je algoritmicky velmi jednoduché a proto se hojně využívá v oblasti výpočetní techniky (uzávorkovaný infixní je převeden na bezzávorkový prefixní tvar).

Příklad

Následující výroky převed'te z infixního tvaru do prefixního:

a) $p \vee \neg q \rightarrow q$

Prefixní tvar: $\rightarrow \vee p \neg q q$

b) $\neg p \wedge q \vee \neg r$

Prefixní tvar: $\vee \wedge \neg p q \neg r$

c) $[(p \vee q) \wedge (p \rightarrow r) \wedge (q \vee r)] \rightarrow r$

Prefixní tvar: $\rightarrow \wedge \wedge \vee p q \rightarrow p r \vee q r r$

Následující výroky převed'te z prefixního tvaru do infixního:

a) $\rightarrow \wedge p q r$

Infixní tvar: $p \wedge q \rightarrow r$

b) $\wedge \neg \rightarrow p q \rightarrow p r$

Infixní tvar: $\neg(p \rightarrow q) \wedge (p \rightarrow r)$

c) $\wedge \wedge p q \rightarrow p \wedge \neg q r$

Infixní tvar: $p \wedge q \wedge (p \rightarrow \neg q \wedge r)$

Poznámka (Booleovské vyhledávání, bitové operace)

- Při vyhledávání v rozsáhlých zdrojích informací (databázové systémy v knihovnách, www vyhledávání apod.) se k upřesnění dotazu velmi často používají logické spojky (především negace, konjunkce a disjunkce). V této souvislosti se pak mluví o tzv. booleovském, resp. rozšířeném vyhledávání.
- Bitovým řetězcem délky $n \in N$ nazveme výraz $x_1 \dots x_n$, kde $x_i \in \{0,1\}$. Bitový řetězec tak tvoří posloupnost nul a jedniček v celkovém počtu n . Někdy budeme psát bitové řetězce do závorek, např. $(x_1 \dots x_n)$. Symboly x_i nazýváme logické, resp. booleovské proměnné a jsou reprezentovány tzv. bitem (bit = binary digit; zavedl v roce 1946 John Tukey). Zavedené logické spojky tak lze interpretovat jako bitové operace. Podrobněji se budeme této problematice věnovat v odstavci o booleovských funkcích.

Definice (tautologie, kontradikce, logická ekvivalence)

- Každý složený výrok, jehož pravdivostní hodnota je pro libovolné pravdivostní ohodnocení atomických formulí (logických proměnných) rovna 1, nazýváme tautologie.
- Každý složený výrok, jehož pravdivostní hodnota je pro libovolné pravdivostní ohodnocení atomických formulí (logických proměnných) rovna 0, nazýváme kontradikce.
- Řekneme, že (složené) výroky p a q jsou logicky ekvivalentní, píšeme $p \Leftrightarrow q$, jestliže výrok $p \Leftrightarrow q$ je tautologie. (Někdy se logická ekvivalence zapisuje i ve tvaru $p = q$.)

Příklad

Dokažte, že platí následující často využívaná logická ekvivalence:

$$p \rightarrow q \Leftrightarrow \neg p \vee q.$$

Stačí tedy dokázat, že výrok $(p \rightarrow q) \Leftrightarrow (\neg p \vee q)$ je tautologie (tj. pravdivý bez ohledu na ohodnocení atomických výroků p, q). Sestrojme příslušnou pravdivostní tabulku.

| p | q | $p \rightarrow q$ | $\neg p \vee q$ | $(p \rightarrow q) \Leftrightarrow (\neg p \vee q)$ |
|-----|-----|-------------------|-----------------|---|
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

(Poslední sloupec není nutný, neboť stačí ověřit, že sloupce odpovídající hodnotám levé i pravé strany logické ekvivalence nabývají stejné hodnoty.)

Stejně snadno zjistíme (pomocí pravdivostní tabulky), že následující výroky jsou tautologiemi:

$$\neg(p \wedge \neg p) \quad (\text{tzv. vyloučení kontradikce})$$

(Vyloučení kontradikce říká, že nemůže současně platit p a jeho negace.)

$$p \vee \neg p. \quad (\text{zákon vyloučeného třetího})$$

(Zákon vyloučeného třetího říká, že platí p , nebo $\neg p$ (tj. není třetí možnost!) a latinsky se vyjadřuje „tertium non datur“.)

Mezi základní logické ekvivalence, které budeme často využívat, patří následující:

| Logická ekvivalence | Název |
|--|----------------------|
| $p \wedge q \Leftrightarrow q \wedge p$ $p \vee q \Leftrightarrow q \vee p$ | Komutativní zákony |
| $(p \wedge q) \wedge r \Leftrightarrow p \wedge (q \wedge r)$ $(p \vee q) \vee r \Leftrightarrow p \vee (q \vee r)$ | Asociativní zákony |
| $p \wedge (q \vee r) \Leftrightarrow (p \wedge q) \vee (p \wedge r)$ $p \vee (q \wedge r) \Leftrightarrow (p \vee q) \wedge (p \vee r)$ | Distributivní zákony |
| $p \wedge p \Leftrightarrow p$ $p \vee p \Leftrightarrow p$ | Zákony idempotence |
| $p \wedge (p \vee q) \Leftrightarrow p$ $p \vee (p \wedge q) \Leftrightarrow p$ | Zákony absorpce |
| $\neg(\neg p) \Leftrightarrow p$ | Zákon dvojité negace |
| $\neg(p \wedge q) \Leftrightarrow \neg p \vee \neg q$ $\neg(p \vee q) \Leftrightarrow \neg p \wedge \neg q$ | De Morganovy zákony |

Jak již bylo řečeno, lze k důkazu logických ekvivalencí využít pravdivostní tabulku. Další možností je provádět vhodné „úpravy“ daného výroku pomocí již známých (dokázaných) logických ekvivalencí. Tento postup demonstruje následující příklad.

Příklad

Dokažte, že výroky $p \rightarrow q$ a $\neg q \rightarrow \neg p$ jsou logicky ekvivalentní.

Jednotlivé kroky důkazu mohou vypadat například takto:

| | |
|--|---|
| $p \rightarrow q \Leftrightarrow \neg p \vee q$ | ... logická ekvivalence z předcházejícího příkladu, |
| $\Leftrightarrow q \vee \neg p$ | ... komutativní zákon, |
| $\Leftrightarrow \neg\neg(q \vee \neg p)$ | ... zákon dvojité negace, |
| $\Leftrightarrow \neg(\neg q \wedge \neg\neg p)$ | ... De Morganovy zákony, |
| $\Leftrightarrow \neg(\neg q \wedge p)$ | ... zákon dvojité negace, |
| $\Leftrightarrow \neg(\neg q) \vee (\neg p)$ | ... De Morganovy zákony, |
| $\Leftrightarrow \neg q \rightarrow \neg p$ | ... logická ekvivalence z předcházejícího příkladu. |

Poznámka

Snadno zjistíme, že existuje celkem $2^4 = 16$ různých logických spojek, kterými lze „spojit“ dva výroky (= počet všech pravdivostních tabulek obsahující dva atomické výroky, tj. mající 4 řádky). Ovšem, zdaleka ne všechny spojky se běžně používají a proto se zmíníme pouze o následujících:

Vylučovací (alternativní) nebo

Nechť p, q jsou výroky, potom symbolem $p \oplus q$ označujeme tzv. vylučovací vztah výroků p, q , tj. výrok „platí buď p , nebo q “. V přirozeném jazyce tak symbolu \oplus odpovídá vylučovací (alternativní) význam spojky „nebo“. Pravdivostní hodnota „vylučovacího nebo“ je dána následující pravdivostní tabulkou.

| p | q | $p \oplus q$ |
|-----|-----|--------------|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Výrok $p \oplus q$ čteme např. „ p exkluzivní or q “, resp. „ p aut q “, „ p vylučovací nebo q “. Způsob čtení není ustálen (spojka bývá často označována jako xor).

Shefferův symbol

Nechť p, q jsou výroky, potom symbolem $p \uparrow q$ (někdy i $p | q$) označujeme tzv. Shefferovu funkci, tj. negaci konjunkce výroků p, q , tedy výrok „neplatí současně p a q “. Pravdivostní hodnota Shefferovy funkce je dána následující pravdivostní tabulkou.

| p | q | $p \uparrow q$ |
|-----|-----|----------------|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Výrok $p \uparrow q$ čteme např. „ p negativní and q “, „ p sheffer q “. Způsob čtení není ustálen (spojka bývá často označována jako nand).

Pierceův symbol

Nechť p, q jsou výroky, potom symbolem $p \downarrow q$ označujeme tzv. Pierceovu funkci, tj. negaci disjunkce výroků p, q , tedy výrok „neplatí p nebo q “. Pravdivostní hodnota Pierceovy funkce je dána následující pravdivostní tabulkou.

| p | q | $p \downarrow q$ |
|-----|-----|------------------|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

Výrok $p \downarrow q$ čteme „ p negativní or q “, resp. „ p pierce q “. Způsob čtení není ustálen (spojka bývá často označována jako nor).

Jako samostatné cvičení přenecháváme důkaz, že platí následující logické ekvivalence (důkaz proveďte pomocí pravdivostních tabulek):

$$p \oplus q \Leftrightarrow \neg(p \leftrightarrow q),$$

$$p \uparrow q \Leftrightarrow \neg(p \wedge q),$$

$$p \downarrow q \Leftrightarrow \neg(p \vee q).$$

Přirozeně nyní vzniká otázka, zda výběr používaných logických spojek nějak omezuje „vyjadřovací“ možnosti výroků. Přesnější formulaci vystihuje níže zavedený pojem úplný systém logických spojek.

Definice

Řekneme, že množina logických spojek S tvoří úplný systém logických spojek, jestliže každý výrok lze vyjádřit pomocí logických spojek množiny S .

Snadno lze ukázat, že kterákoliv z následujících množin

$$S_1 = \{\neg, \wedge, \vee\}, \quad S_2 = \{\neg, \wedge\}, \quad S_3 = \{\neg, \vee\}, \quad S_4 = \{\neg, \rightarrow\}, \quad S_5 = \{\uparrow\}, \quad S_6 = \{\downarrow\}$$

tvoří úplný systém logických spojek.

Návod - pomocí logických spojek dané množiny vyjádřete všechny ostatní spojky. Např.

$$\neg p \Leftrightarrow p \downarrow p, \quad p \vee q \Leftrightarrow (p \downarrow q) \downarrow (p \downarrow q), \quad p \wedge q \Leftrightarrow (p \downarrow p) \downarrow (q \downarrow q),$$

$$\neg p \Leftrightarrow p \uparrow p, \quad p \vee q \Leftrightarrow (p \uparrow p) \uparrow (q \uparrow q), \quad p \wedge q \Leftrightarrow (p \uparrow q) \uparrow (p \uparrow q).$$

Množiny

$$S_7 = \{\wedge, \vee\}, \quad S_8 = \{\neg, \leftrightarrow\}$$

úplný systém netvoří (dokažte).

Důsledkem skutečnosti, že množina $\{\neg, \wedge, \vee\}$ tvoří úplný systém, je možnost vyjádřit libovolný výrok pouze pomocí negace, konjunkce a disjunkce, navíc tvar takových výroků lze „standardizovat“ v následujícím smyslu:

- Řekneme, že výrok je disjunktivním normálním tvaru (DNF), jestliže má tvar

$$p_1 \vee p_2 \vee \dots \vee p_k,$$

kde každé $p_i, i = 1, \dots, k$ je konjunkce literálů (literál = atomický výrok nebo jeho negace).

- Řekneme, že výrok je konjunktivním normálním tvaru (CNF), jestliže má tvar

$$p_1 \wedge p_2 \wedge \dots \wedge p_l,$$

kde každé $p_i, i = 1, \dots, l$ je disjunkce literálů.

Indukcí (dle složitosti) lze dokázat, že platí:

Každý výrok lze převést na disjunktivní nebo konjunktivní normální tvar.

Způsob nalezení DNF a CNF popisují následující algoritmy.

Algoritmus DNF

(Převáděný výrok není kontradikce a obsahuje n atomických výroků p_1, \dots, p_n .)

1. Sestrojte pravdivostní tabulku (obsahuje 2^n řádků).
2. Ke každému řádku pravdivostní tabulky, kde výrok nabývá hodnoty 1, přiřaďte konjunktivní člen

$$x_1 \wedge x_2 \wedge \dots \wedge x_n,$$

kde $x_i = p_i$, jestliže je v daném řádku $p_i = 1$ a $x_i = \neg p_i$ v ostatních případech.

3. DNF zapište jako disjunkci všech výše vytvořených konjunktivních členů.

Algoritmus CNF

(Převáděný výrok není tautologie a obsahuje n atomických výroků p_1, \dots, p_n .)

1. Sestrojte pravdivostní tabulku.
2. Ke každému řádku pravdivostní tabulky, kde výrok nabývá hodnoty 0, přiřaďte disjunktivní člen

$$x_1 \vee x_2 \vee \dots \vee x_n,$$

kde $x_i = p_i$, jestliže je v daném řádku $p_i = 0$ a $x_i = \neg p_i$ v ostatních případech.

3. CNF zapište jako konjunkci všech výše vytvořených disjunktivních členů.

Příklad

Nalezněte (aplikací výše popsaných algoritmů) disjunktivní i konjunktivní normální tvar výroku $((p \wedge \neg q) \rightarrow (p \rightarrow r)) \wedge (\neg q \vee r)$

| p | q | r | $((p \wedge \neg q) \rightarrow (p \rightarrow r)) \wedge (\neg q \vee r)$ | DNF | CNF |
|-----|-----|-----|--|--------------------------------------|-----------------------------|
| 0 | 0 | 0 | 1 | $\neg p \wedge \neg q \wedge \neg r$ | --- |
| 0 | 0 | 1 | 1 | $\neg p \wedge \neg q \wedge r$ | --- |
| 0 | 1 | 0 | 0 | --- | $p \vee \neg q \vee r$ |
| 0 | 1 | 1 | 1 | $\neg p \wedge q \wedge r$ | --- |
| 1 | 0 | 0 | 0 | --- | $\neg p \vee q \vee r$ |
| 1 | 0 | 1 | 1 | $p \wedge \neg q \wedge r$ | --- |
| 1 | 1 | 0 | 0 | --- | $\neg p \vee \neg q \vee r$ |
| 1 | 1 | 1 | 1 | $p \wedge q \wedge r$ | --- |

DNF: $(\neg p \wedge \neg q \wedge \neg r) \vee (\neg p \wedge \neg q \wedge r) \vee (\neg p \wedge q \wedge r) \vee (p \wedge \neg q \wedge r) \vee (p \wedge q \wedge r)$

CNF: $(p \vee \neg q \vee r) \wedge (\neg p \vee q \vee r) \wedge (\neg p \vee \neg q \vee r)$

2.2 Booleovské funkce

Při analyzování funkce celé řady zařízení často zjistíme, že fungují tzv. dvoustavově. Činnost těchto zařízení lze popsat pomocí booleovských funkcí, tj. funkcí, které nabývají, stejně jako jejich proměnné, pouze dvou hodnot, nejčastěji označovaných 1, 0. Interpretace těchto hodnot je závislá na kontextu a může být například „vede – nevede“ u reléových a elektronických obvodů, resp. „funguje – porucha“ v oblasti spolehlivosti. Tím se dostáváme k pojmům booleovská operace, booleovská proměnná a booleovská funkce.

Definice

1. Proměnnou, která může nabývat pouze hodnoty z množiny $B = \{0,1\}$ nazveme booleovskou proměnnou.
2. Na množině B definujeme booleovské operace[†] sčítání „+“, násobení „ \cdot “ tabulkami:

| | | |
|---|---|---|
| + | 0 | 1 |
| 0 | 0 | 1 |
| 1 | 1 | 1 |

| | | |
|---------|---|---|
| \cdot | 0 | 1 |
| 0 | 0 | 0 |
| 1 | 0 | 1 |

a operaci negace „ $\bar{}$ “ vztahy $\bar{0} = 1, \bar{1} = 0$.

3. Booleovskou funkcí n proměnných nazveme každé zobrazení $f: B^n \rightarrow B$, kde B^n označuje n -tou kartézskou mocninu množiny $B = \{0,1\}$.

Poznámka

- Booleovské proměnné budeme zpravidla označovat malými písmeny z konce abecedy, např. v, w, x, y, z apod. V nutných případech použijeme i dolní indexy, např. x_1, x_2 apod.
- Definiční obor booleovské funkce n proměnných tvoří všechny bitové řetězce délky n , tj. množina $B^n = \{(x_1, \dots, x_n) \mid x_i \in \{0,1\}, 1 \leq i \leq n\}$. Oborem hodnot je množina $B = \{0,1\}$.
- Booleovské operace sčítání, násobení a negace lze přirozeným způsobem rozšířit i na booleovské funkce. Jsou-li $f, g: B^n \rightarrow B$ booleovské funkce, píšeme $f + g, f \cdot g, \bar{f}$, kde

$$(f + g)(x_1, \dots, x_n) = f(x_1, \dots, x_n) + g(x_1, \dots, x_n),$$

[†] Booleovské operace jsou vyhodnocovány v následujícím pořadí: negace, násobení a sčítání. Například $xy + z$ je vyhodnoceno jako $(xy) + z$, nikoliv $x(y + z)$.

$$(f \cdot g)(x_1, \dots, x_n) = f(x_1, \dots, x_n) \cdot g(x_1, \dots, x_n),$$

$$\overline{f}(x_1, \dots, x_n) = \overline{f(x_1, \dots, x_n)}.$$

(Symbol pro booleovský součin „ \cdot “ budeme vynechávat. V případě proměnných píšeme xy místo $x \cdot y$ a v případě booleovských funkcí fg místo $f \cdot g$.)

- Symbolem **0** označujeme booleovskou funkci, která nabývá v celém definičním oboru hodnotu 0. Analogicky symbolem **1** označujeme booleovskou funkci, která nabývá pouze hodnotu 1.

(Pozor: **0** \neq 0 a **1** \neq 1, neboť tučné symboly označují funkce, nikoliv prvky množiny B .)

Každá booleovská funkce je zřejmě určena hodnotami, které nabývá na prvcích svého definičního oboru B^n a lze ji proto zadat pomocí tabulky funkčních hodnot. Tato tabulka obsahuje 2^n řádků, které reprezentují všechny bitové řetězce z B^n a jim odpovídající funkční hodnoty. Snadno zjistíme, že existuje 2^{2^n} různých booleovských funkcí n proměnných (= počet bitových řetězců délky 2^n). Přehled všech booleovských funkcí dvou proměnných obsahuje následující tabulka:

| x | y | f_1 | f_2 | f_3 | f_4 | f_5 | f_6 | f_7 | f_8 | f_9 | f_{10} | f_{11} | f_{12} | f_{13} | f_{14} | f_{15} | f_{16} |
|-----|-----|-------|-------|-------|-------|-------|-------|-------|-------|-------|----------|----------|----------|----------|----------|----------|----------|
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 |

Další možností jak zadat booleovskou funkci jsou booleovské výrazy[†]. Ty definujeme jako výrazy vzniklé konečným počtem užití následujících dvou pravidel:

1. Každá booleovská proměnná je booleovský výraz.
2. Jsou-li B_1, B_2 booleovské výrazy, potom

$$\left. \begin{array}{l} \overline{B_1} \\ B_1 + B_2 \\ B_1 \cdot B_2 \end{array} \right\} \text{ jsou booleovské výrazy.}$$

[†] V další části uvidíme, že oba způsoby (tabulka hodnot, booleovské výrazy) zadání booleovských funkcí jsou ekvivalentní a ke každé tabulce hodnot tak lze sestavit odpovídající booleovský výraz.

Snadno zjistíme, že všechny booleovské funkce dvou proměnných, které jsme označili f_1, \dots, f_{16} , lze reprezentovat následujícími booleovskými výrazy:

$$\begin{array}{lll}
 f_1(x, y) = 1, & f_2(x, y) = x + y, & f_3(x, y) = x + \bar{y}, \\
 f_4(x, y) = \bar{x} + y, & f_5(x, y) = \overline{xy}, & f_6(x, y) = x, \\
 f_7(x, y) = y, & f_8(x, y) = xy + \bar{x}\bar{y}, & f_9(x, y) = x\bar{y} + \bar{x}y, \\
 f_{10}(x, y) = \bar{y}, & f_{11}(x, y) = \bar{x}, & f_{12}(x, y) = xy, \\
 f_{13}(x, y) = x\bar{y}, & f_{14}(x, y) = \bar{x}y, & f_{15}(x, y) = \overline{x + y}, \\
 f_{16}(x, y) = 0.
 \end{array}$$

Pro funkce f_5, f_9, f_{15} se navíc vžila speciální symbolika i název:

$f_5(x, y) = x \uparrow y$... Shefferova funkce (negativní and, NAND),

$f_9(x, y) = x \oplus y$... vylučující nebo (XOR),

$f_{15}(x, y) = x \downarrow y$... Peirceova funkce (negativní or, NOR).

Důležité je si uvědomit, že dva formálně odlišné booleovské výrazy mohou reprezentovat stejnou funkci. Tím se dostáváme k pojmu ekvivalence booleovských funkcí (resp. výrazů). Řekneme, že dvě booleovské funkce $f, g : B^n \rightarrow B$ jsou ekvivalentní, píšeme $f = g$, jestliže obě funkce nabývají pro stejné hodnoty proměnných stejné funkční hodnoty, tj. $\forall (x_1, \dots, x_n) \in B^n$ platí $f(x_1, \dots, x_n) = g(x_1, \dots, x_n)$.

Příklad

Ukažte, že pro funkce $f(x, y, z) = xy + (x + y)\bar{z} + y$ a $g(x, y, z) = x\bar{z} + y$ platí $f = g$.

Zřejmě stačí sestavit tabulku jejich hodnot.

| x | y | z | xy | $x + y$ | $(x + y)\bar{z}$ | $f(x, y, z)$ | $x\bar{z}$ | $g(x, y, z)$ |
|-----|-----|-----|------|---------|------------------|--------------|------------|--------------|
| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Jelikož sloupce obsahující funkční hodnoty $f(x, y, z)$ a $g(x, y, z)$ jsou shodné, platí $f = g$.

Početní pravidla platná pro booleovské proměnné i funkce jsou v některých ohledech odlišná od „běžných“ početních pravidel a jsou shrnuta v následující tabulce:

| Booleovské proměnné | Booleovské funkce | Název |
|--|--|----------------------|
| $(x + y) + z = x + (y + z)$ $(xy)z = x(yz)$ | $(f + g) + h = f + (g + h)$ $(fg)h = f(gh)$ | Asociativní zákony |
| $x + y = y + x$ $xy = yx$ | $f + g = g + f$ $fg = gf$ | Komutativní zákony |
| $(x + y)(x + z) = x + yz$ $x(y + z) = xy + xz$ | $(f + g)(f + h) = f + gh$ $f(g + h) = fg + fh$ | Distributivní zákony |
| $\overline{\overline{x + y}} = \overline{x} \overline{y}$ $\overline{xy} = \overline{x} + \overline{y}$ | $\overline{\overline{f + g}} = \overline{f} \overline{g}$ $\overline{fg} = \overline{f} + \overline{g}$ | DeMorganovy zákony |
| $x + xy = x$ $x(x + y) = x$ | $f + fg = f$ $f(f + g) = f$ | Zákony absorpce |
| $\overline{\overline{x}} = x$ | $\overline{\overline{f}} = f$ | Zákon dvojité negace |
| $x + x = x$ $xx = x$ | $f + x = x$ $xx = x$ | Zákony idempotence |
| $x + 0 = x$ $x \cdot 1 = x$ | $f + 0 = f$ $f \cdot 1 = f$ | Zákony identity |
| $x + \overline{x} = 1$ $x \overline{x} = 0$ | $f + \overline{f} = 1$ $f \overline{f} = 0$ | Zákony inverze |
| $x + 1 = 1$ $x \cdot 0 = 0$ | $f + 1 = 1$ $f \cdot 0 = 0$ | Zákony dominance |

(Platnost uvedených pravidel lze snadno prověřit např. pomocí tabulky hodnot. Jako cvičení proveďte platnost prvního distributivního zákona, obě DeMorganova pravidla a oba zákony absorpce.)

V další části tohoto odstavce se budeme věnovat následujícím otázkám:

1. Mějme booleovskou funkci zadanou tabulkou funkčních hodnot. Jak lze nalézt jí odpovídající booleovský výraz?
2. Které booleovské operace (tj. booleovské funkce dvou proměnných) tvoří funkčně úplný systém, tj. umožňují vyjádřit libovolnou booleovskou funkci?

Odpověď na první otázku dají následující tvrzení. Nejprve však zavedeme nezbytnou terminologii. Označme $f(x_1, \dots, x_n)$ booleovskou funkci n proměnných. Literálem nazveme libovolnou booleovskou proměnnou x_i nebo její negaci \bar{x}_i , kde $1 \leq i \leq n$. Řekneme, že $f(x_1, \dots, x_n)$ je v úplném disjunktivním normálním tvaru (označíme DNF), jestliže je vyjádřena ve tvaru součtu součinů $y_1 \cdot \dots \cdot y_n$, kde y_i je literál odpovídající proměnné x_i . Analogicky řekneme, že $f(x_1, \dots, x_n)$ je v úplném konjunktivním normálním tvaru (CNF), jestliže je vyjádřena ve tvaru součinu součtů $(y_1 + \dots + y_n)$, kde y_i je literál odpovídající proměnné x_i .

Tvrzení

- Každou booleovskou funkci, která není identicky rovna 0, lze vyjádřit v úplném disjunktivním normálním tvaru.
- Každou booleovskou funkci, která není identicky rovna 1, lze vyjádřit v úplném konjunktivním normálním tvaru.

(Pravdivost těchto tvrzení ihned vyplývá z následujících algoritmů konstrukce DNF a CNF. Formální důkaz neuvádíme.)

Konstrukce DNF

1. K dané booleovské funkci $f(x_1, \dots, x_n) \neq 0$ sestavte tabulku hodnot.
2. Každému řádku, ve kterém je $f(x_1, \dots, x_n) = 1$ přiřaďte součin literálů $y_1 \cdot \dots \cdot y_n$ (tzv. miniterm), kde $y_i = x_i$, jestliže je v daném řádku $x_i = 1$ a $y_i = \bar{x}_i$ v ostatních případech (tj. $x_i = 0$).
3. DNF je součtem všech minitermů vytvořených v druhém bodu.

Konstrukce CNF

1. K dané booleovské funkci $f(x_1, \dots, x_n) \neq 1$ sestavte tabulku hodnot.
2. Každému řádku, ve kterém je $f(x_1, \dots, x_n) = 0$ přiřaďte součet literálů $(y_1 + \dots + y_n)$ (tzv. maxterm), kde $y_i = x_i$, jestliže je v daném řádku $x_i = 0$ a $y_i = \bar{x}_i$ v ostatních případech (tj. $x_i = 1$).
3. CNF je součinem všech maxtermů vytvořených v druhém bodu.

Příklad

Nalezněte DNF a CNF funkce $f(x, y, z) = ((\bar{x} \uparrow y) \oplus (y \downarrow z)) + yz$.

| x | y | z | $f(x, y, z)$ | DNF | CNF |
|-----|-----|-----|--------------|---------------------|-------------------|
| 1 | 1 | 1 | 1 | $x y z$ | --- |
| 1 | 1 | 0 | 1 | $x y \bar{z}$ | --- |
| 1 | 0 | 1 | 1 | $x \bar{y} z$ | --- |
| 1 | 0 | 0 | 0 | --- | $\bar{x} + y + z$ |
| 0 | 1 | 1 | 1 | $\bar{x} y z$ | --- |
| 0 | 1 | 0 | 0 | --- | $x + \bar{y} + z$ |
| 0 | 0 | 1 | 1 | $\bar{x} \bar{y} z$ | --- |
| 0 | 0 | 0 | 0 | --- | $x + y + z$ |

DNF je součtem minitermů ze sloupce označeného DNF, tj.

$$f(x, y, z) = x y z + x y \bar{z} + x \bar{y} z + \bar{x} y z + \bar{x} \bar{y} z.$$

CNF je součinem maxtermů ze sloupce označeného CNF, tj.

$$f(x, y, z) = (\bar{x} + y + z)(x + \bar{y} + z)(x + y + z).$$

Poznámka

Pokud již máme k dispozici např. DNF, lze CNF snadno nalézt užitím dvojité negace a početních pravidel, zejména DeMorganových. Je-li $f(x, y) = x \bar{y} + \bar{x} y$, dostáváme

$$\overline{f(x, y)} = \overline{x \bar{y} + \bar{x} y} = (\bar{x} + y)(x + \bar{y}) = \bar{x} \bar{y} + x y,$$

tedy

$$f(x, y) = \overline{\overline{f(x, y)}} = \overline{\bar{x} \bar{y} + x y} = (x + y)(\bar{x} + \bar{y}).$$

Nyní krátce k druhé otázce. Víme, že počet všech booleovských funkcí n proměnných je dán vztahem 2^{2^n} . Konkrétní představu o „extrémně“ rychlém nárůstu počtu booleovských funkcí v závislosti na počtu proměnných, si lze udělat z následující tabulky. Přirozeně se tak dostáváme k otázce existence funkčně úplných systémů (množin) booleovských operací. Systém booleovských operací nazveme funkčně úplný, jestliže každou booleovskou funkci lze vyjádřit pomocí booleovského výrazu využívajícího pouze operace daného systému.

| Počet proměnných | Počet booleovských funkcí |
|------------------|---|
| 1 | 4 |
| 2 | 16 |
| 3 | 256 |
| 4 | 65 536 |
| 5 | 4 294 967 296 |
| 6 | 18 446 744 073 709 551 616 |
| 7 | 340 282 366 920 938 463 463 374 607 431 768 211 456 |

Jako důsledek existence DNF, resp. CNF ihned dostáváme platnost tvrzení, že množina booleovských operací $\{\neg, +, \cdot\}$ je funkčně úplná. Dále lze snadno ukázat, že každá z následujících množin

$$S_1 = \{\neg, +\}, S_2 = \{\neg, \cdot\}, S_3 = \{\uparrow\}, S_4 = \{\downarrow\}$$

tvoří také funkčně úplný systém (množina $\{+, \cdot\}$ není funkčně úplná!).

K důkazu funkční úplnosti zřejmě postačuje vyjádřit operace $\neg, +, \cdot$ (víme, že tvoří funkčně úplný systém) pomocí operací daných systémů. V případě S_1 , resp. S_2 stačí využít zákon dvojité negace a DeMorganova pravidla, tj.

$$xy = \overline{\overline{x} + \overline{y}}, \text{ resp. } x + y = \overline{\overline{x} \cdot \overline{y}},$$

v případě S_3 využijeme vztahy (dokažte je)

$$\overline{x} = x \uparrow x, \quad x + y = (x \uparrow x) \uparrow (y \uparrow y), \quad xy = (x \uparrow y) \uparrow (x \uparrow y)$$

a v případě S_4 vztahy

$$\overline{x} = x \downarrow x, \quad x + y = (x \downarrow y) \downarrow (x \downarrow y), \quad xy = (x \downarrow x) \downarrow (y \downarrow y).$$

Řekneme, že booleovská funkce je v minimálním tvaru, jestliže je vyjádřena ve tvaru součtu minimálního počtu sčítanců, kde každý sčítanec je součinem minimálního počtu literálů. (Na rozdíl od DNF tak nemusí být v každém sčítanci minimálního tvaru obsaženy literály reprezentující všechny proměnné.)

Karnaughova mapa (Maurice Karnaugh, 1924)

- V závislosti na počtu proměnných použijte některé z následujících schémat:

| | yz | $\bar{y}z$ | $\bar{y}\bar{z}$ | $y\bar{z}$ |
|------------------|------|------------|------------------|------------|
| wx | | | | |
| $\bar{w}x$ | | | | |
| $\bar{w}\bar{x}$ | | | | |
| $w\bar{x}$ | | | | |

V odpovídajícím schématu označíme políčka reprezentující jednotlivé sčítance DNF.

- Pomocí níže uvedených typů pokrývacích bloků nalezneme pokrytí označených políček, které splňuje podmínky:

- 50

| Počet proměnných | Typy pokrývacích bloků | | | | | | | | |
|---------------------|------------------------|-----|-----|-----|-----|-----|-----|-----|-----|
| 2 | 1×1 | 1×2 | 2×1 | 2×2 | | | | | |
| 3 | 1×1 | 1×2 | 1×4 | 2×1 | 2×2 | 2×4 | | | |
| 4 | 1×1 | 1×2 | 1×4 | 2×1 | 2×2 | 2×4 | 4×1 | 4×2 | 4×4 |

Pozor, z hlediska pokrývání jsou poslední a první sloupec, resp. řádek sousední. Tuto skutečnost zachycují následující ukázky:

| | | | | |
|-----------|------|------------|------------------|------------|
| | yz | $\bar{y}z$ | $\bar{y}\bar{z}$ | $y\bar{z}$ |
| x | √ | | | √ |
| \bar{x} | | | | |

| | | | | |
|-----------|------|------------|------------------|------------|
| | yz | $\bar{y}z$ | $\bar{y}\bar{z}$ | $y\bar{z}$ |
| x | √ | | | √ |
| \bar{x} | √ | | | √ |

| | | | | |
|------------------|------|------------|------------------|------------|
| | yz | $\bar{y}z$ | $\bar{y}\bar{z}$ | $y\bar{z}$ |
| wx | √ | | | √ |
| $\bar{w}x$ | √ | | | √ |
| $\bar{w}\bar{x}$ | √ | | | √ |
| $w\bar{x}$ | √ | | | √ |

4. Hledaný minimální tvar je součtem součinů literálů, které jsou společné jednotlivým pokrývacím blokům.

Použití Karnaughovy mapy ilustruje následující příklad.

Příklad

Pomocí Karnaughovy mapy minimalizujte následující booleovské funkce:

a) $f_1(x, y) = xy + x\bar{y} + \bar{x}y$,

b) $f_2(w, x, y, z) = wxyz + wxy\bar{z} + \bar{w}xyz + \bar{w}xy\bar{z} + \bar{w}\bar{x}\bar{y}z + \bar{w}\bar{x}y\bar{z} + \bar{w}x\bar{y}\bar{z} + w\bar{x}y\bar{z}$.

ad a) Vzhledem k tomu, že $f_1(x, y)$ je v DNF, sestojíme příslušnou Karnaughovu mapu:

| | | |
|-----------|-----|-----------|
| | y | \bar{y} |
| x | √ | √ |
| \bar{x} | √ | |

(Políčka označená symbolem √ reprezentují jednotlivé sčítance DNF.)

Dle pravidel uvedených ve 3. bodě nalezneme nejlepší pokrytí. Jelikož jde o funkci dvou proměnných, využijeme pokrývací struktury typu: 1×1, 1×2, 2×1, 2×2. V tomto případě je nejlepší pokrytí zobrazeno následujícím schématem.

| | | |
|-----------|-----|-----------|
| | y | \bar{y} |
| x | √ | √ |
| \bar{x} | √ | |

Na závěr určíme literály společné jednotlivým pokrývacím blokům. V případě bloku typu 1×2 je společný pouze literál x a pokrývacímu bloku typu 2×1 je společný pouze literál y . Jako hledaný minimalizovaný tvar tak dostáváme $f_1(x, y) = x + y$.

ad b) Jelikož funkce $f_2(w, x, y, z)$ je v DNF, sestrojíme příslušnou Karnaughovu mapu a nalezneme nejlepší pokrytí.

| | yz | $\bar{y}z$ | $\bar{y}\bar{z}$ | $y\bar{z}$ |
|------------------|--------------|--------------|------------------|--------------|
| wx | \checkmark | | | \checkmark |
| $\bar{w}x$ | \checkmark | | | \checkmark |
| $\bar{w}\bar{x}$ | | \checkmark | \checkmark | \checkmark |
| $w\bar{x}$ | | | | \checkmark |

(Pozor, první a poslední sloupec jsou sousední, proto využíváme pokrývací blok 2×2 .)

Pokrývacímu bloku 1×2 odpovídá součin $\bar{w}\bar{x}\bar{y}$, bloku 2×2 součin xy a bloku 4×1 součin $y\bar{z}$. Jako hledaný minimalizovaný tvar dostáváme

$$f_2(w, x, y, z) = \bar{w}\bar{x}\bar{y} + xy + y\bar{z}.$$

Při počítačové minimalizaci booleovských funkcí více proměnných se běžně používá následující algoritmus.

Quine-McCluskeyův algoritmus (Willard Quine, 1908; Edward McCluskey, 1929)

- Booleovskou funkci $f(x_1, \dots, x_n)$ vyjádřete v DNF. Literály každého sčítance zapisujte v pevně daném pořadí y_1, \dots, y_n a přiřaďte jim jejich bitovou reprezentaci, tj. bitový řetězec $(b_1 \dots b_n)$, kde $b_i = 1$, jestliže $y_i = x_i$ a $b_i = 0$, jestliže $y_i = \bar{x}_i$.
- Sestrojte a vyplňte tabulku dle následujících pravidel:
 - Sloupec 1: Sloupec vyplňte očíslovaným seznamem sčítanců DNF a jejich bitovou reprezentací. Uvedený seznam sčítanců запиšte v sestupném pořadí daném počtem jedniček vyskytujících se v jejich bitové reprezentaci. Například sčítanec $x_1\bar{x}_2x_3x_4$ je reprezentován bitovým řetězcem (1011) a předchází sčítanec $x_1x_2\bar{x}_3\bar{x}_4$ reprezentovaný bitovým řetězcem (1100).
 - Sloupec 2: Do tohoto sloupce запиšte všechny členy, které vzniknou tzv. redukcí, tj. součtem (využijte distributivní zákony) těch dvojic členů z prvního sloupce, které se

liší v právě jednom literálu a přiřaďte jim rozšířenou bitovou reprezentaci $(b_1 \dots b_n)$, kde $b_i = 1$, jestliže $y_i = x_i$, $b_i = 0$, jestliže $y_i = \bar{x}_i$ a $b_i = "-"$ (pomlčka), jestliže proměnná x_i není ve vzniklém členu zastoupena. Například z dvojice sčítanců $x_1 x_2 x_3 \bar{x}_4$ a $x_1 x_2 \bar{x}_3 \bar{x}_4$ v prvním sloupci vznikne redukcí člen $x_1 x_2 \bar{x}_4$ s rozšířenou bitovou reprezentací 11–0.

- Sloupec 3 – atd.: Členy dalších sloupců vzniknou opět redukcí, ovšem jako součty dvojic členů z bezprostředně předcházejícího sloupce. Sčítané členy se musí lišit v právě jedné pozici. Všem takto vzniklým členům přiřadíme jejich rozšířenou bitovou reprezentaci. Ve vytváření dalších sloupců pokračujte, dokud existují dvojice, které lze redukovat (sčítat).
3. Sestrojte tabulku jejíž sloupce jsou postupně označeny všemi členy DNF a řádky jsou označeny všemi členy z tabulky vytvořené v bodu 2, které nebyly využity v procesu redukce. Postupně projděte jednotlivá políčka tabulky a označte pole v i -tém řádku a j -tém sloupci, pokud všechny literály z označení i -tého řádku jsou součástí označení j -tého sloupce.
 4. Nalezněte pokrytí, tj. minimální počet řádků, ve kterých jsou označeny všechny sloupce tabulky zkonstruované ve 3. bodu. Minimalizovaný tvar funkce $f(x_1, \dots, x_n)$ je pak součtem označení řádků tvořících libovolné pokrytí – nemusí být určeno jednoznačně. (Toto je „nejobtížnější“ část Quine-McCluskeyova algoritmu, která je obvykle realizována metodou nazývanou backtracking.)

Použití Quine-McCluskeyova algoritmu ilustruje následující příklad.

Příklad

Minimalizujte

$$f(w, x, y, z) = \bar{w} \bar{x} \bar{y} \bar{z} + \bar{w} \bar{x} \bar{y} z + \bar{w} \bar{x} y \bar{z} + \bar{w} \bar{x} y z + \bar{w} x y \bar{z} + \bar{w} x y z + w x y \bar{z} + w x y z.$$

Minimalizovaná funkce je v DNF a bitová reprezentace jednotlivých členů je následující:

$$\begin{array}{lll} \bar{w} \bar{x} \bar{y} \bar{z} \dots (0000), & \bar{w} \bar{x} \bar{y} z \dots (0001), & \bar{w} \bar{x} y \bar{z} \dots (0010), \\ \bar{w} \bar{x} y z \dots (0011), & \bar{w} x y \bar{z} \dots (0110), & \bar{w} x y z \dots (0111), \\ w x y \bar{z} \dots (1110), & w x y z \dots (1111). \end{array}$$

Tabulka sestavená dle druhého bodu má tvar

| | | | | | | | | |
|---|--------------------------------|--------|-------|-------------------------|--------|-----------|------------------|--------|
| 1 | $wxyz$ | (1111) | (1,2) | wxy | (111-) | (1,2,3,4) | xy | (-11-) |
| 2 | $wxy\bar{z}$ | (1110) | (1,3) | xyz | (-111) | (3,4,5,6) | $\bar{w}y$ | (0-1-) |
| 3 | $\bar{w}xyz$ | (0111) | (2,4) | $xy\bar{z}$ | (-110) | (5,6,7,8) | $\bar{w}\bar{x}$ | (00--) |
| 4 | $\bar{w}xy\bar{z}$ | (0110) | (3,4) | $\bar{w}xy$ | (011-) | | | |
| 5 | $\bar{w}\bar{x}yz$ | (0011) | (3,5) | $\bar{w}yz$ | (0-11) | | | |
| 6 | $\bar{w}\bar{x}y\bar{z}$ | (0010) | (4,6) | $\bar{w}y\bar{z}$ | (0-10) | | | |
| 7 | $\bar{w}\bar{x}\bar{y}z$ | (0001) | (5,6) | $\bar{w}\bar{x}y$ | (001-) | | | |
| 8 | $\bar{w}\bar{x}\bar{y}\bar{z}$ | (0000) | (5,7) | $\bar{w}\bar{x}z$ | (00-1) | | | |
| | | | (6,8) | $\bar{w}\bar{x}\bar{z}$ | 00-0 | | | |

První sloupec obsahuje všechny členy DNF (seřazeny vzestupně dle počtu negací), jejich pořadí a bitovou reprezentaci. Druhý sloupec obsahuje členy vzniklé redukcí dvojic členů (lišících se v právě jedné pozici; čísla v závorkách udávají pořadová čísla) z prvního sloupce a jejich rozšířenou bitovou reprezentaci. Třetí sloupec obsahuje členy, které vznikly redukcí dvojic členů ze druhého sloupce a jejich rozšířenou bitovou reprezentaci. Tabulka obsahuje tři sloupce, neboť členy tohoto sloupce již nelze redukovat.

Tabulka sestavená dle třetího bodu má tvar

| | $wxyz$ | $wxy\bar{z}$ | $\bar{w}xyz$ | $\bar{w}xy\bar{z}$ | $\bar{w}\bar{x}yz$ | $\bar{w}\bar{x}y\bar{z}$ | $\bar{w}\bar{x}\bar{y}z$ | $\bar{w}\bar{x}\bar{y}\bar{z}$ |
|------------------|--------|--------------|--------------|--------------------|--------------------|--------------------------|--------------------------|--------------------------------|
| xy | | | | | √ | √ | √ | √ |
| $\bar{w}y$ | | | √ | √ | √ | √ | | |
| $\bar{w}\bar{x}$ | √ | √ | √ | √ | | | | |

Odtud již dostáváme hledaný minimalizovaný tvar $f(w, x, y, z) = \bar{w}\bar{x} + xy$.

2.3 Základy predikátové logiky

Nyní se opět vrátíme k základům matematické logiky a řekneme si něco o predikátové logice. Připomeňme, že tvrzení typu $x \leq 27$ nebo $x - y = 0$ apod. nejsou výroky, neboť obsahují proměnné a není tak možné určit jejich pravdivostní hodnotu. Teprve v okamžiku, kdy za proměnné dosadíme konkrétní hodnoty z tzv. univerza (v uvedených příkladech to je množina reálných čísel), jde o výrok. Tím se dostáváme k pojmu predikát. Predikátem budeme rozumět tvrzení, které obsahuje proměnné a u kterého lze po dosazení konkrétních hodnot proměnných určit pravdivostní hodnotu. V případech, kdy se tvrzení týká pouze jedné proměnné (např. $x \leq 27$), mluvíme o jednomístném predikátu, který budeme obvykle značit symbolem $P(x)$. V obecném případě pak mluvíme o vícemístném (n místném) predikátu, který budeme značit $P(x_1, \dots, x_n)$. Predikáty lze považovat za jakési stavební kameny, pomocí kterých budou tvořeny výrazy nazývané formule predikátové logiky.

Příklad

- Za univerzum zvolme množinu všech celých čísel Z a definujme jednomístný predikát $P(x)$ jako tvrzení „ x je sudé“. Zřejmě tak $P(15)$ je nepravdivý výrok, neboť číslo 15 není sudé a naopak $P(-68)$ je pravdivý výrok.
- Jako univerzum nyní zvolme množinu reálných čísel R a definujme třímístný predikát $P(x, y, z)$ vztahem $x - y = z$. V tomto případě je $P(0, 0, 1)$ nepravdivý výrok a $P(-2, 3, -5)$ pravdivý.

Stejným způsobem jako v případě výrokové logiky, lze i při konstrukci formulí predikátové logiky využívat logické spojky (zavedené v odstavci o výrokové logice). Platí:

Jsou-li $P(x_1, \dots, x_n)$, $Q(x_1, \dots, x_m)$ predikáty, potom výrazy

$$\left. \begin{array}{l} \neg P(x_1, \dots, x_n) \\ P(x_1, \dots, x_n) \wedge Q(x_1, \dots, x_m) \\ P(x_1, \dots, x_n) \vee Q(x_1, \dots, x_m) \\ P(x_1, \dots, x_n) \rightarrow Q(x_1, \dots, x_m) \\ P(x_1, \dots, x_n) \leftrightarrow Q(x_1, \dots, x_m) \end{array} \right\} \text{ jsou formule predikátové logiky.}$$

(Priority logických spojek jsou stejné jako v případě výrokové logiky.)

Dalším způsobem jak lze, kromě dosazení konkrétních hodnot za proměnné, vytvořit z formulí predikátové logiky výroky, je použití tzv. kvantifikátorů neboli kvantorů. V matematice (ale i v běžném životě) se často používají tvrzení, typu: každý prvek uvažovaného univerza má jistou vlastnost (popsanou predikátem), resp. existuje alespoň jeden prvek univerza s danou vlastností. Například každý absolvent TUL složil úspěšně zkoušku z některého cizího jazyka, existuje alespoň jeden student TUL, který složil všechny předepsané zkoušky z matematiky v řádném termínu, apod. Tato tvrzení odpovídají také formulím predikátové logiky a v symbolické podobě se zapisují právě pomocí kvantifikátorů, kde symbolem \forall budeme označovat tzv. obecný kvantifikátor (nazývaný také velký) a symbolem \exists existenční (malý) kvantifikátor. Způsob jejich použití je následující:

$\forall x P(x)$ označuje výrok „Pro každý prvek univerza platí $P(x)$.“

$\exists x P(x)$ označuje výrok „Existuje prvek univerza, pro který platí $P(x)$.“

$\exists! x P(x)$ označuje výrok „Existuje právě jeden prvek univerza, pro který platí $P(x)$.“

Pro zvýšení přehlednosti se někdy využívají závorky, zvláště u formulí, které obsahují logické spojky. Píšeme např.: $\forall x [P(x) \rightarrow \neg Q(x)]$, resp. $\exists x [P(x) \wedge Q(x)]$ apod.

Poznamenejme dále, že proměnné, kterým byla přiřazena hodnota, resp. jsou kvantifikovány, se nazývají vázané proměnné a ostatní se nazývají volné (mluvíme o vázaném, resp. volném výskytu proměnné). Z formule predikátové logiky se tak stává výrok pouze tehdy, pokud všechny proměnné, které se v ní vyskytují jsou vázané.

Poznámka

- Pro negaci obecného kvantifikátoru platí logická ekvivalence:

$$\neg \forall x P(x) \Leftrightarrow \exists x \neg P(x)$$

která vyjadřuje, že tvrzení „není pravda, že pro každé x platí $P(x)$ “ je stejné (logicky ekvivalentní) s tvrzením „existuje x , pro které neplatí $P(x)$ “.

- Pro negaci existenčního kvantifikátoru platí logická ekvivalence

$$\neg \exists x P(x) \Leftrightarrow \forall x \neg P(x),$$

která vyjadřuje, že tvrzení „neexistuje x , pro které platí $P(x)$ “ je stejné (logicky ekvivalentní) s tvrzením „pro každé x neplatí $P(x)$ “.

- Pro negaci kvantifikátoru jedinečné existence $\exists!$ platí logická ekvivalence

$$\neg \exists! x P(x) \Leftrightarrow \neg \exists x P(x) \vee \exists y \exists z [(y \neq z) \wedge P(y) \wedge P(z)].$$

(Pravé straně odpovídají dvě možnosti – neexistuje žádné x pro které platí $P(x)$, nebo existují alespoň dva různé prvky, pro které platí $P(x)$.)

- V případech, kdy univerzum je konečná množina, např. $U = \{x_1, \dots, x_n\}$, zřejmě platí následující logické ekvivalence:

$$\forall x P(x) \Leftrightarrow P(x_1) \wedge \dots \wedge P(x_n),$$

$$\exists x P(x) \Leftrightarrow P(x_1) \vee P(x_2) \vee \dots \vee P(x_n).$$

Při popisu složitějších vlastností je často nutné použít i více než jeden kvantifikátor, přičemž jejich pořadí hraje důležitou roli. Základní představu o významu pořadí kvantifikátorů si lze udělat z následující tabulky a příkladu.

| Tvrzení | Pravdivé | Nepravdivé |
|--|---|---|
| $\forall x \forall y P(x, y)$ $\forall y \forall x P(x, y)$ | Pro každou dvojici x, y je $P(x, y)$ pravdivé. | Existuje dvojice x, y , pro kterou je $P(x, y)$ nepravdivé. |
| $\forall x \exists y P(x, y)$ | Ke každému x existuje y takové, že $P(x, y)$ je pravdivé. | Existuje x takové, že $P(x, y)$ je nepravdivé pro každé y . |
| $\exists x \forall y P(x, y)$ | Existuje x takové, že je $P(x, y)$ pravdivé pro každé y . | Ke každému x existuje y takové, že $P(x, y)$ je nepravdivé. |
| $\exists x \exists y P(x, y)$ $\exists y \exists x P(x, y)$ | Existuje dvojice x, y , pro kterou je $P(x, y)$ pravdivé. | Pro každou dvojici x, y je $P(x, y)$ nepravdivé. |

Příklad

- a) Vysvětlete rozdíl mezi tvrzením $\forall x \exists y \quad x + y = 0$ a $\exists y \forall x \quad x + y = 0$, kde $x, y \in \mathbb{R}$

První tvrzení je pravdivé a vyjadřuje skutečnost, že ke každému reálnému číslu x existuje opačné číslo y .

Druhé tvrzení je nepravdivé, neboť by muselo existovat reálné číslo y (nezávislé na volbě x), které je opačné ke každému reálnému číslu x .

- b) „Upravte“ tvrzení: $\neg \forall x [\exists y \forall z P(x, y, z) \wedge \exists z \forall y P(x, y, z)]$.

$$\neg \forall x [\exists y \forall z P(x, y, z) \wedge \exists z \forall y P(x, y, z)] \Leftrightarrow \exists x [\forall y \exists z \neg P(x, y, z) \vee \forall z \exists y \neg P(x, y, z)].$$

Kapitola 3. Základy teorie grafů

Pojem grafu

V této kapitole se budeme zabývat významným oborem diskrétní matematiky, a to teorií grafů. Grafy v této teorii jsou něco zcela jiného než grafy funkcí, se kterými mají jen společný název odvozený z řeckého slova „grafein“ (psáti). Grafů užíváme tehdy, potřebujeme-li znázornit spojení mezi dvojicemi objektů. Bývá to například u schématického znázornění železniční sítě, které se uvádí v jízdních řádech. Na obr. 1 (str. 59) je takové znázornění určité části železniční sítě na severu Čech, na obr. 2 (str. 59) je síť pražského metra. Dalším příkladem jsou strukturní vzorce v chemii. V těchto vzorcích jsou jednotlivé atomy, z nichž se skládá molekula sloučeniny, znázorněny svými chemickými značkami a vazba mezi dvěma atomy je vyznačena úsečkou. Na obr. 3 (str. 60) je strukturní vzorec butanu. V moderní době pronikají grafy i do společenských věd, například do sociologie a psychologie. Můžeme třeba znázornit jednotlivé žáky ve třídě kroužky a dva kroužky spojit úsečkou tehdy, je-li mezi příslušnými žáky kamarádský vztah (musíme si ovšem napřed vymezit, co rozumíme kamarádským vztahem). Dostaneme opět graf. Podobné grafy mohou řídicím pracovníkům pomoci sestavit vhodným způsobem pracovní skupiny na pracovišti.

Ve všech případech máme nějaké objekty, které znázorňujeme body, kroužky či jinak a nazýváme uzly, a mezi nimi hrany, které znázorňujeme úsečkami nebo oblouky. Graf ovšem není takovýto nákres, ale struktura tvořená těmi uzly a hranami. Budeme zde mluvit o tzv. neorientovaných grafech a slovem graf budeme rozumět graf neorientovaný. Nyní již lze zformulovat obecnou definici grafu.

Neorientovaný graf je uspořádaná dvojice (U, H) , kde U je konečná množina prvků zvaných uzly grafu (někdy také vrcholy grafu) a H je podmnožina množiny všech dvouprvkových podmnožin U . Prvky množiny H se nazývají hrany grafu.

Obvykle se grafy označují velkými písmeny latinské abecedy, tedy píšeme např. $G = (U, H)$. Je-li $u \in U, v \in U, \{u, v\} \in H$, říkáme, že uzly u, v jsou spojeny hranou $\{u, v\}$. Hrana ovšem může být také označena jedním malým latinským písmenem, tedy např. $h = \{u, v\}$. Jsou-li uzly u, v spojeny hranou h , říkáme o nich také, že jsou sousední nebo adjacentní. Hrana h a kterýkoliv z uzlů u, v jsou spolu incidentní a uzly u, v jsou koncové uzly hrany h .

Uvědomme si, že H je množina určitých dvouprvkových podmnožin množiny U , tedy uspořádaných dvojic prvků množiny U . Z neuspořádané dvojice $\{u, v\}$ lze vytvořit dvě uspořádané dvojice (u, v) , (v, u) a kdybychom k dané množině H vytvořili množinu takovýchto uspořádaných dvojic, byla by to symetrická binární relace na množině U . A obráceně ke každé symetrické relaci na U lze takto přiřadit neorientovaný graf. Mohli bychom říci, že neorientovaný graf $G = (U, H)$ je symetrická binární relace na množině U , a nemuseli bychom zavádět termín graf. Přesto však, protože teorie grafů je poněkud jinak zaměřena než teorie relací, rozlišujeme mezi oběma pojmy a jejich teoriemi.

Způsoby zadávání grafu

Jistě jste už pochopili, že graf není geometrický náčrtek, ale je to určitá abstraktní struktura daná množinou uzlů a údaji o tom, které dvojice uzlů jsou spojeny hranou. Nechceme-li zrovna kreslit obrázek, můžeme graf popsat některým z následujících způsobů:

- seznamem uzlů a jejich následníků,
- maticí sousednosti (adjecence),
- maticí incidence.

Pro další výklad předpokládejme, že uvažovaný graf $G = (U, H)$ má n uzlů (tj. $U = \{u_1, \dots, u_n\}$) a m hran (tj. $H = \{h_1, \dots, h_m\}$). Nyní podrobněji k jednotlivým způsobům.

Seznam uzlů a následníků

Tento způsob zadávání grafu, se nejčastěji používá při počítačové implementaci grafů a spočívá v uvedení (spojového) seznamu všech uzlů a jejich následníků. Například graf z obrázku 4 lze zadat následujícím seznamem:

$$u_1 \longrightarrow \{u_2, u_3, u_5\},$$

$$u_2 \longrightarrow \{u_1, u_3\},$$

$$u_3 \longrightarrow \{u_1, u_2, u_4\},$$

$$u_4 \longrightarrow \{u_3, u_5\},$$

$$u_5 \longrightarrow \{u_1, u_4, u_6\},$$

$$u_6 \longrightarrow \{u_5\}.$$

Matice sousednosti

Jde o čtvercovou matici $A_G = (a_{i,j})_{i,j=1}^n$ řádu n , tj. mající n řádků a sloupců (reprezentují uzly), definovanou vztahy:

$a_{i,j} = 1$, jestliže $\{u_i, u_j\} \in H$, tj. uzly u_i a u_j jsou spojeny hranou,
0, v ostatních případech (uzly u_i a u_j nejsou spojeny hranou).

Například grafu z obr. 4 odpovídá následující matice sousednosti:

$$A_G = \begin{pmatrix} 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Matice incidence

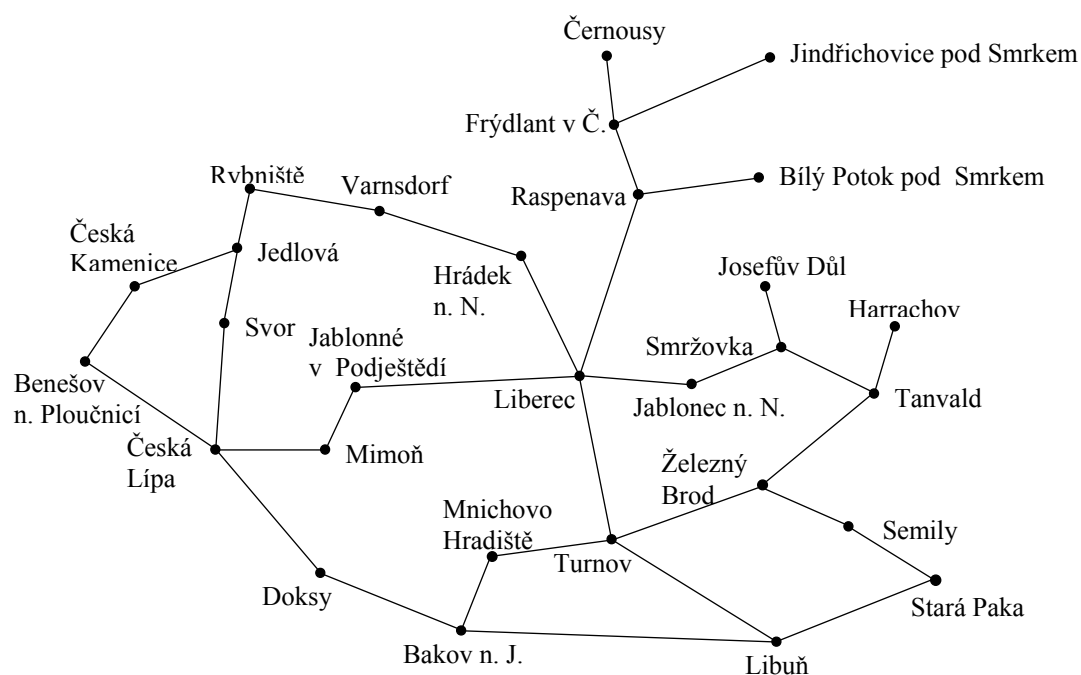
Jde o matici $M_G = (m_{i,j})_{i,j=1}^{n,m}$ řádu $n \times m$, tj. mající n řádků (reprezentují uzly grafu) a m sloupců (reprezentují hrany grafu), definovanou vztahy:

$m_{i,j} = 1$, jestliže $u_i \in h_j$, tj. hrana h_j obsahuje uzel u_i ,
0, v ostatních případech.

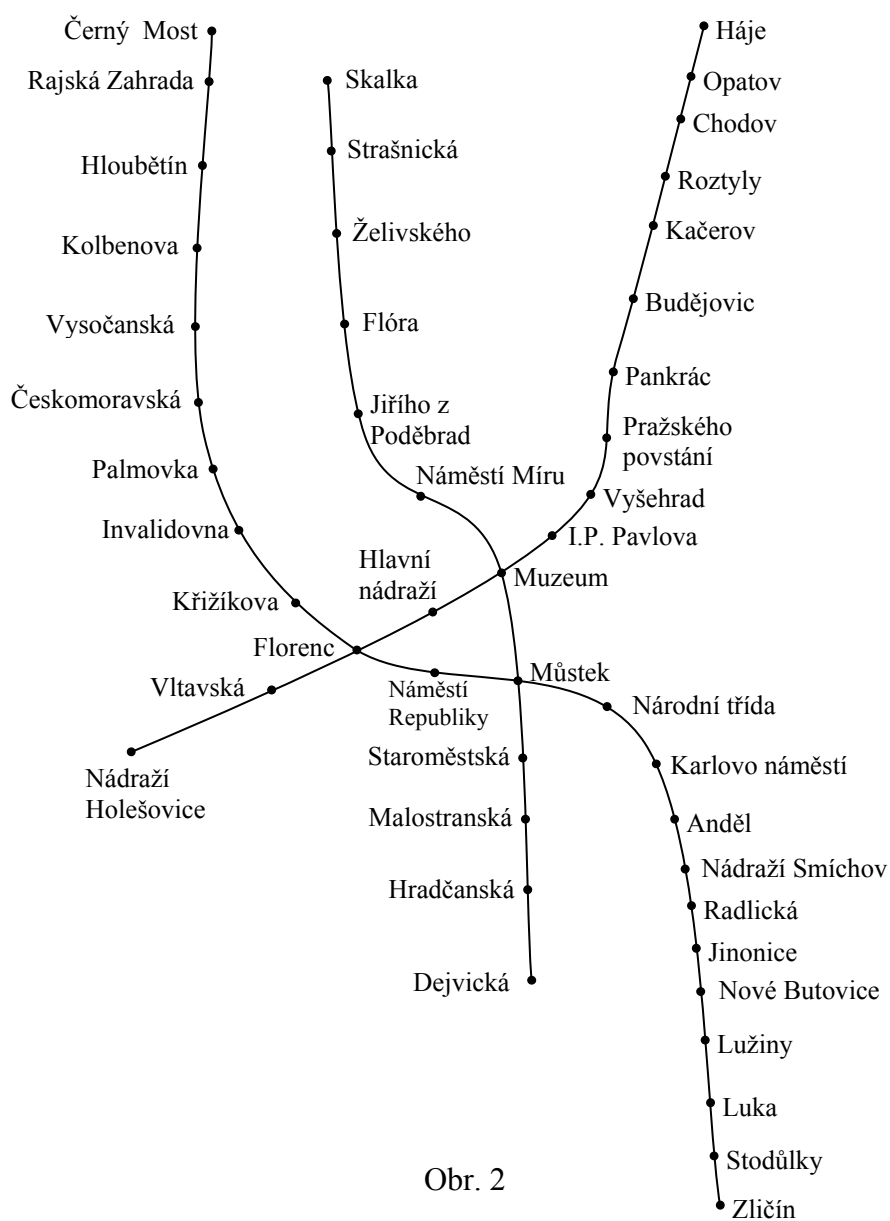
Známemu grafu z obr. 4 tak odpovídá následující matice incidence:

$$M_G = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

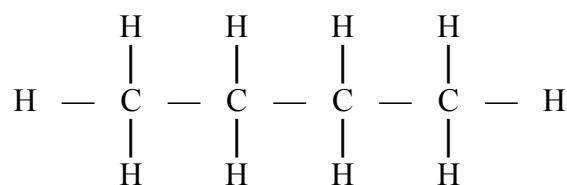
Na závěr poznamenejme, že v některých aplikacích se lze setkat i s grafy, jejichž uzly jsou spojeny více než jednou hranou nebo je uzel spojen hranou sám se sebou (tzv. smyčka). Např. u strukturních vzorců v chemii se dvojná či trojná vazba atomů znázorní příslušným počtem úseček. Na obr. 5 vidíme strukturní vzorec kyanovodíku, kde mezi atomy dusíku a uhlíku je trojná vazba. Těmito grafy se však zde zabývat nebudeme.



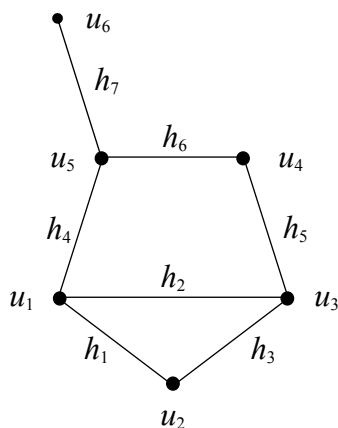
Obr. 1



Obr. 2



Obr. 3



Obr. 4



Obr. 5

Stupně uzlů v grafu

Důležitým pojmem teorie grafů je tzv. stupeň uzlu, definovaný následovně:

Nechť $G = (U, H)$ je graf a u jeho libovolný uzel (tj. $u \in U$). Potom počet hran grafu G incidentních s uzlem u nazýváme stupeň uzlu u v grafu G a značíme $d_G(u)$ nebo jednodušeji $d(u)$.

Např. v grafu železniční trati nám stupeň uzlu ukazuje, kolika směry může z dané stanice vyjet vlak. Na obr. 1 vidíme, že z Liberce můžeme vyjet pěti směry. V grafu pražského metra na obr. 2 máme tři uzly stupně 4; jsou to přestupní stanice Muzeum, Můstek a Florenc. Stupeň 1 mají konečné stanice Dejvická, Skalka, Černý Most, Nádraží Holešovice a Háje. Ostatní stanice jsou uzly stupně 2.

Minimální stupeň uzlu v grafu G se značí $d(u)$, maximální $\Delta(G)$. Sčítáme-li stupně všech uzlů grafu G , počítáme vlastně jeho hrany, ale každou dvakrát, zvlášť u každého jejího koncového uzlu.

Platí tedy:

$$\sum_{u \in U} d_G(u) = 2 \cdot |H|,$$

Znamená to ovšem také, že $\sum_{u \in U} d_G(u)$ je vždy sudé číslo. A z toho plyne toto tvrzení:

V každém grafu je počet uzlů lichého stupně vždy sudý.

Zavedme nyní pojem grafové posloupnosti:

Nechť $G = (U, H)$ je graf o n uzlech. Grafová posloupnost příslušná grafu G je konečná nerostoucí posloupnost n čísel, v níž se stupeň každého uzlu grafu G vyskytuje tolikrát, kolikrát se vyskytuje v grafu G .

Jde tedy o posloupnost stupňů grafu G , které jsou seřazeny do nerostoucí posloupnosti. Přirozeně vzniká otázka, zda každá posloupnost obsahující n nezáporných čísel s_1, s_2, \dots, s_n , přičemž $s_1 \geq s_2 \geq \dots \geq s_n$ tvoří grafovou posloupnost příslušnou nějakému grafu.

Je-li dána posloupnost o popsanych vlastnostech, můžeme určitým algoritmem určit, zda existuje graf G , pro nějž je grafovou posloupností. Autorem algoritmu je český matematik Václav Jaromír Havel.

Mějme tedy posloupnost (s_1, s_2, \dots, s_n) . Vynechme z ní první člen s_1 . Dále od každého členu s_j pro $2 \leq j \leq s_1 + 1$ odečteme číslo 1. Pokud by to nešlo, protože by bylo méně než $s_1 + 1$ členů, je odpověď na položenou otázku NE. Jinak dostaneme novou konečnou posloupnost, která má pouze $n - 1$ členů. Skládá-li se ze samých nul, je odpověď na otázku ANO. Obsahuje-li záporné číslo, je odpověď NE. Jinak pokud nová posloupnost není neklesající, patřičně ji upravíme změnou uspořádání. Nato opakujeme algoritmus s novou posloupností.

Příklad.

Rozhodněte, zda následující posloupnosti tvoří grafovou posloupnost:

a) (5, 4, 3, 3, 3, 3, 2, 2, 2, 1)

Vynecháním prvního členu a odčítáním od dalších dostaneme posloupnost

$$(3, 2, 2, 2, 2, 2, 2, 2, 1).$$

Zde nemusíme měnit uspořádání a můžeme pokračovat. Dostáváme tak

$$(1, 1, 1, 2, 2, 2, 2, 1).$$

a po změně uspořádání

$$(2, 2, 2, 2, 1, 1, 1, 1).$$

Další pokračování vypadá následovně:

$$(2, 2, 2, 2, 1, 1, 1, 1) \rightarrow (1, 1, 2, 1, 1, 1, 1, 1) \rightarrow (2, 1, 1, 1, 1, 1, 1, 1) \rightarrow (0, 0, 1, 1, 1, 1, 1, 1) \rightarrow \\ \rightarrow (1, 1, 1, 1, 0, 0, 0, 0) \rightarrow (0, 1, 1, 0, 0, 0, 0, 0) \rightarrow (1, 1, 0, 0, 0, 0, 0, 0) \rightarrow (0, 0, 0, 0, 0, 0, 0, 0).$$

Odpověď je ANO.

b) (5, 5, 4, 3, 2, 1)

Průběh algoritmu vypadá takto:

$$(5, 5, 4, 3, 2, 1) \rightarrow (4, 3, 2, 1, 0) \rightarrow (2, 1, 0, -1).$$

Odpověď je NE.

V dalším výkladu využijeme tyto speciální termíny:

- Je-li $d_G(u) = 0$, uzel u se nazývá izolovaný.
- Existuje-li číslo k takové, že $d_G(u) = k$ pro každé $u \in U$, graf G se nazývá pravidelný graf stupně k .

Závěrem ještě ukážeme, že v každém grafu se alespoň u dvou uzlů jejich stupně rovnají.

Předpokládejme, že tomu tak není. Potom stupně tvoří n různých čísel. Je zřejmé, že $0 \leq d_G(u) \leq n-1$ pro každé $u \in U$, tedy všechna čísla $0, 1, \dots, n-1$ by se vyskytla jako stupně uzlů. Pokud však existuje uzel u_1 stupně 0, nemůže existovat uzel u_2 stupně $n-1$; žádný uzel totiž není spojen s uzlem u_1 ani sám se sebou, tedy jeho stupeň může být nejvýše $n-2$. To je ovšem spor s předpokladem a tvrzení je dokázáno.

Podgraf grafu

Podobně jako v první kapitole jsme zavedli pojem podmnožiny, zde zavedeme pojem podgrafu daného grafu.

Buďte $G = (U, H)$, $G_0 = (U_0, H_0)$ dva grafy. Je-li $U_0 \subseteq U$ a $H_0 \subseteq H$, pak graf G_0 nazýváme podgrafem grafu G .

Rozumí se samozřejmě, že H_0 je nějaká množina dvouprvkových podmnožin množiny H . To však nemusíme výslovně uvádět, protože je to řečeno už tím, že G_0 je graf.

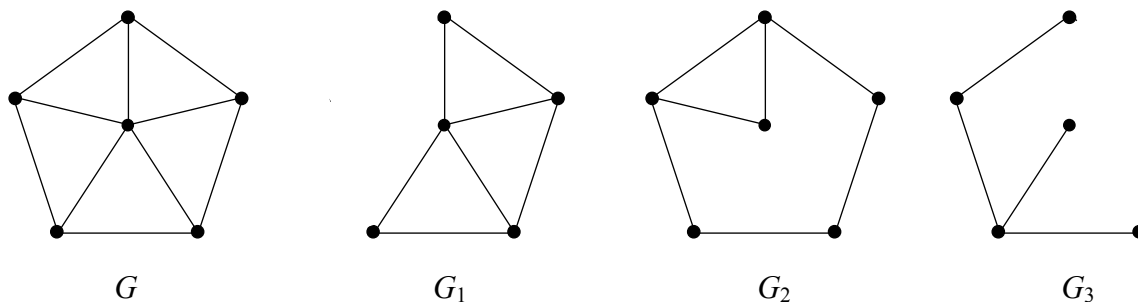
Existují dva významné typy podgrafů, a to indukovaný podgraf a faktor.

Budiž graf $G_0 = (U_0, H_0)$ podgrafem grafu $G = (U, H)$. Je-li H_0 množina všech dvouprvkových podmnožin množiny U_0 , které patří do množiny H , pak G_0 se nazývá indukovaný podgraf grafu G , případně podgraf grafu G indukovaný množinou U_0 .

Je-li dána libovolná podmnožina U_0 množiny U , je indukovaný podgraf $G_0 = (U_0, H_0)$ jednoznačně určen. Jeho množina uzlů je U_0 , jeho množina hran je množina všech dvouprvkových podmnožin množiny U_0 , která jsou hranami grafu G . Kdybychom chtěli graf G_0 kreslit, nakreslili bychom G , vyznačili v něm uzly množiny U_0 a vedli všechny hrany mezi těmito uzly, které jsou v grafu G .

Nyní přistoupíme k definici faktoru. Budiž graf $G_0 = (U_0, H_0)$ podgrafem grafu $G = (U, H)$. Je-li $U_0 = U$, pak graf G_0 se nazývá faktor grafu G .

Na obr. 6 vidíme graf G (takovýto graf bývá nazýván kolo) a jeho tři podgrafy G_1, G_2, G_3 . Graf G_1 je indukovaný podgraf grafu G , graf G_2 je faktor grafu G , graf G_3 je podgraf grafu G , který není ani jeho indukovaný podgraf, ani jeho faktor.



Obr. 6

Souvislost grafu

Vezměme si opět schématické znázornění železniční sítě pomocí grafu. Co to znamená, že máme spojení ze stanice a do stanice b ? Znamená to, že máme možnost se pohybovat vlakem postupně po jednotlivých hranách, vždy od jednoho koncového uzlu ke druhému, takovým způsobem, že začneme v uzlu a a skončíme v uzlu b . Tak dostáváme definici spojení v grafu:

Nechť a, b jsou dva uzly grafu $G = (U, H)$. Spojení z uzlu a do uzlu b v grafu G je konečná posloupnost tvaru

$$a = u_0, h_1, u_1, h_2, \dots, u_{k-1}, h_k, u_k = b,$$

kde $u_i \in U$ pro $i = 0, 1, \dots, k$ a $h_i = \{u_{i-1}, u_i\} \in H$ pro $i = 1, \dots, k$.

Nyní na základě pojmu spojení vyslovíme další definici:

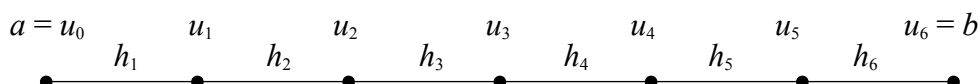
Říkáme, že uzly a, b grafu G spolu souvisí, jestliže v G existuje spojení z uzlu a do uzlu b .

Pro spojení se někdy také užívá termínu sled. Poznamenejme, že můžeme mít také spojení složené z jediného uzlu a žádné hrany; tedy každý uzel souvisí sám se sebou.

Ve spojení se obecně mohou uzly i hrany opakovat. Máme však určité speciální typy spojení. Spojení z uzlu a do uzlu b v grafu G , v němž se každá hrana grafu G vyskytuje nejvýše jednou, se nazývá tah z uzlu a do uzlu b v grafu G . Jestliže v tom případě $a = b$, pak se takový tah nazývá uzavřený, jinak se nazývá otevřený.

Otevřený tah z uzlu a do uzlu b v grafu G , v němž se každý uzel grafu G vyskytuje nejvýše jednou, se nazývá cesta z uzlu a do uzlu b v grafu G . Počet hran cesty C z a do b se nazývá délka cesty C .

Na obr. 7 vidíme cestu z a do b délky 6. Graf tvořený uzly a hranami jediné cesty se někdy nazývá had, častěji se mu však přímo říká cesta. Je to graf, který se buď skládá z jediného izolovaného uzlu, $a = b$, nebo je takový, že v něm existují dva uzly stupně 1 (uzly a, b) a všechny ostatní uzly mají stupeň 2.



Obr. 7

Když si najdeme vhodné spojení na železnici, bývá toto spojení cestou. Nestává se, že bychom projížděli několikrát toutéž stanicí. Toho se samozřejmě dá docílit jednoduše, vynecháváním některých úseků. Platí totiž věta.

Existuje-li v grafu G spojení z uzlu a do uzlu b , existuje v něm i cesta z uzlu a do uzlu b .

Vraťme se k pojmu souvisení dvou uzlů. Je to zřejmě binární relace σ na množině uzlů U . Jak jsme se výše zmínili, každý uzel souvisí sám se sebou. Tedy $(u, u) \in \sigma$ pro každé $u \in U$ a relace σ je reflexivní. Máme-li spojení z a do b a napíšeme-li je v opačném pořadí, dostaneme spojení z b do a . Tedy z $(a, b) \in \sigma$ vyplývá $(b, a) \in \sigma$ a relace σ je symetrická. Máme-li spojení z a do b a další spojení z b do c v grafu G , můžeme to druhé spojení umístit za to první a dostaneme spojení z a do c . Tedy z $(a, b) \in \sigma$ a $(b, c) \in \sigma$ vyplývá $(a, c) \in \sigma$ a relace σ je rovněž tranzitivní.

Vidíme, že σ je relace ekvivalence na množině U . Existuje tedy rozklad $\mathfrak{R} = \{V_1, \dots, V_k\}$ množiny U takový, že dva uzly u, v jsou ve stejné třídě rozkladu \mathfrak{R} právě tehdy, jestliže spolu souvisí. A víme, že ke každé podmnožině množiny U existuje podgraf grafu G indukovaný tou podmnožinou. Pro $i = 1, \dots, r$ tedy můžeme symbolem G_i označit podgraf grafu G indukovaný množinou V_i . Grafům G_1, \dots, G_r budeme říkat komponenty souvislosti (nebo krátce komponenty) grafu G .

Samozřejmě dvě různé komponenty grafu G nemají společný uzel (a tím méně společnou hranu). Každá hrana grafu G spojuje uzly téže komponenty. Pokud graf G znázorníme nákresem, bude nám v něm každá komponenta připadat jako samostatný graf, zcela oddělený od ostatních. Uzly různých komponent prostě spolu nesouvisí.

Jestliže graf G obsahuje pouze jedinou komponentu, znamená to, že $(u, v) \in \sigma$ pro každá $u \in U, v \in U$, tedy že σ je univerzální relace na U . Libovolné dva uzly grafu G spolu souvisí. Pak říkáme, že G je souvislý graf.

Železniční síť České republiky (nepočítáme-li lanovky) tvoří souvislý graf a tedy z kterékoliv naší železniční stanice se vlakem dostaneme do kterékoli jiné.

Některé speciální typy grafů

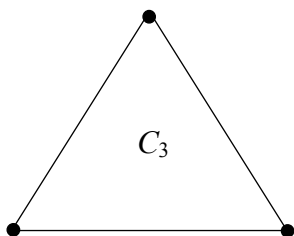
V otázkách týkajících se souvislosti grafů hrají důležitou roli grafy zvané kružnice.

Kružnice C_n délky n je konečný souvislý graf, který je pravidelný a obsahuje právě n uzlů stupně 2.

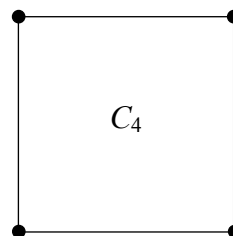
Dala by se rovněž popsat jako graf tvořený uzly a hranami uzavřeného tahu, v němž se každý uzel vyskytuje nejvýše jednou. Délka kružnice je počet jejích hran. Na obr. 8, 9, 10, 11 vidíme kružnice C_3 , C_4 , C_5 , C_6 délek postupně 3, 4, 5, 6.

Významný je podgraf grafu, který je kružnicí. Náleží-li hrana h grafu G některému takovému podgrafu, nazývá se cyklická; jinak se nazývá acyklická neboli most. Odstraníme-li acyklickou hranu $h = \{u, v\}$ z grafu G , pak v grafu G' , který takto vznikne, uzly u, v spolu nesouvisí (vysvětlte proč) a počet komponent je o jednu větší než u grafu G . Z komponenty, která obsahovala hranu h , vzniknou dvě komponenty, které se nazývají břehy mostu h . Speciálně, je-li graf G souvislý, pak se odstraněním acyklické hrany stane nesouvislým, zatímco v případě, kdy odstraníme cyklickou hranu, toto nenastane.

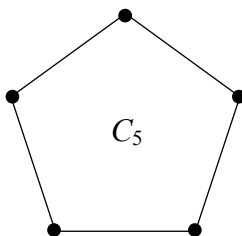
Souvislý graf, který neobsahuje žádnou kružnici jako podgraf, a jehož všechny hrany jsou tudíž acyklické, se nazývá strom; bude mu věnován samostatný paragraf.



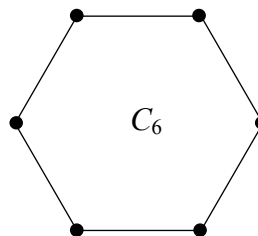
Obr. 8



Obr. 9



Obr. 10



Obr. 11

Graf o n uzlech, v němž každé dva různé uzly jsou spolu spojeny hranou, se nazývá úplný graf K_n . Na obr. 12, 13, 14, 15, 16 jsou úplné grafy K_1, K_2, K_3, K_4, K_5 . Graf K_n je vždy pravidelný graf stupně $n-1$.

K_1



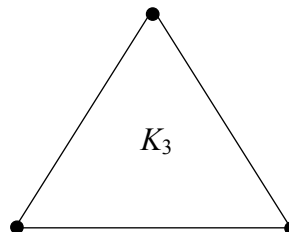
Obr. 12

K_2



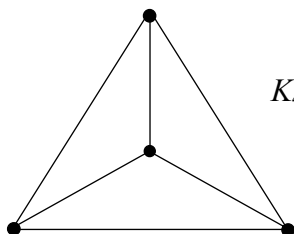
Obr. 13

K_3



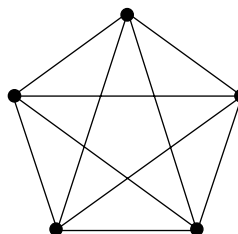
Obr. 14

K_4



Obr. 15

K_5



Obr. 16

Jestliže graf G neobsahuje žádnou kružnici liché délky jako svůj podgraf a má aspoň dva uzly, nazývá se sudý nebo bipartitní. Vždy existuje rozklad $\{U, V\}$ jeho množiny uzlů U na dvě třídy takový, že každá hrana grafu G spojuje uzel z V s uzlem z W . Speciálně každý strom je sudý graf.

Jestliže v takovém grafu každý uzel z V je spojen hranou s každým uzlem z W a platí $|V| = m, |W| = n$, pak takový graf se nazývá úplný sudý graf $K_{m,n}$.

Poznamenejme, že úplný sudý graf není úplný graf. Úplný sudý graf $K_{m,n}$ pro $m = 1, n \geq 2$ je strom a nazývá se hvězda o n hranách.

Isomorfismus grafů

Pojem isomorfismu grafů je analogický pojmu shodnosti v geometrii.

Nechť $G_1 = (U_1, H_1)$, $G_2 = (U_2, H_2)$ jsou dva grafy a necht' existuje prosté zobrazení $\varphi: U_1 \rightarrow U_2$ takové, že pro libovolné dva uzly u, v grafu G_1 je $\{\varphi(u), \varphi(v)\} \in H_2$ právě tehdy, je-li $\{u, v\} \in H_1$. Potom φ se nazývá isomorfní zobrazení grafu G_1 na graf G_2 .

Existuje-li isomorfní zobrazení grafu G_1 na graf G_2 , říkáme, že grafy G_1, G_2 jsou spolu isomorfní a píšeme $G_1 \cong G_2$.

Znázorníme-li grafy G_1, G_2 nákresy, pak při $G_1 \cong G_2$ můžeme říci, že tyto nákresy „vypadají stejně“. Máme například $K_{1,1} \cong K_2$, $K_3 \cong C_3$, $K_{2,2} \cong C_4$.

Vzdálenost uzlů v grafu

Podobně jako třeba v euklidovském prostoru máme definovanu vzdálenost dvou bodů, můžeme i v souvislém grafu definovat vzdálenost mezi dvěma uzly.

Budiž G konečný souvislý graf, buďtež u, v dva jeho uzly. Minimum délek cest z u do v v grafu G se nazývá vzdálenost uzlů u, v v grafu G a značí se $d_G(u, v)$.

Poněvadž graf G je souvislý, pro libovolné uzly u, v musí existovat spojení z u do v , a tedy i cesta z u do v . Ze všech takových cest musí některá mít nejmenší délku, takže $d_G(u, v)$ je vždy definováno.

Podobně jako vzdálenosti mezi jinými matematickými objekty, i vzdálenost $d_G(u, v)$ splňuje takzvané metrické axiomy, a je to tudíž metrika na množině U . Jsou to tyto axiomy:

- $d_G(u, v) = 0$ právě tehdy, jestliže $u = v$,
- $d_G(u, v) \geq 0$ pro každé $u, v \in U$, (nezápornost)
- $d_G(u, v) = d_G(v, u)$ pro každé $u, v \in U$, (symetrie)
- $d_G(u, v) + d_G(v, w) \geq d_G(u, w)$ pro každé $u, v, w \in U$. (trojúhelníková nerovnost)

O metrikách na množinách se více dovíte v jiných matematických předmětech. Z pojmů majících vztah k vzdálenosti je nejdůležitější průměr a poloměr grafu.

Průměr grafu G (značíme $d(G)$) je maximum vzdáleností $d_G(u, v)$ mezi jednotlivými dvojicemi uzlů u, v grafu G .

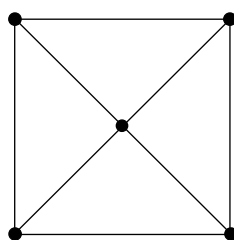
Z grafů o n uzlech platí $d(G) = 1$ pouze pro $G \cong K_n$ a $d(G) = n - 1$ pouze pro cestu (hada) délky $n - 1$. Jinak ovšem vždy platí $1 \leq d(G) \leq n - 1$.

Trochu složitěji se definuje poloměr grafu G . Nejprve je třeba definovat excentricitu uzlu u v grafu G .

Nechť G je konečný souvislý graf, nechť u je jeho uzel. Excentricita uzlu u v grafu G (značíme $e_G(u)$) je maximum vzdáleností $d_G(u, v)$ pro všechny uzly $v \in U - \{u\}$.

Minimum excentricit $e_G(u)$ pro všechny uzly $u \in U$ se nazývá poloměr grafu G a značí se $r(G)$.

Například kolo z obr. 17 má $r(G) = 1$, $d(G) = 2$.



Obr. 17

Vztah mezi poloměrem a průměrem není tak jednoduchý jako u kružnice v geometrii, ale přece je podobný. Platí

$$r(G) \leq d(G) \leq 2 \cdot r(G).$$

Právě u kružnice C_n (v grafovém smyslu) platí

$$r(C_n) = d(C_n) = \left\lfloor \frac{n}{2} \right\rfloor,$$

kde $\lfloor \cdot \rfloor$ označuje funkci dolní celá část.

Uzel $u \in U$ takový, že $e_G(u) = r(G)$, se nazývá centrální uzel grafu G a množina všech centrálních uzlů grafu G se nazývá centrum grafu G .

Centrum grafu se může skládat z jediného uzlu, ale může být také rovno celému U (tento případ nastává například u kružnice C_n a u úplného grafu K_n).

Stromy, kostry grafu

Nechť $T = (U, H)$ je souvislý graf, který neobsahuje žádnou kružnici jako svůj podgraf. Potom T se nazývá strom.

Strom budeme značit písmenem T (z anglického slova „tree“), případně s indexy, např. T_1, T_2 apod.

Představme si, že postupně kreslíme nákres stromu, a to tak, že začneme libovolnou hranou a pak vždy kreslíme takovou hranu, jejíž jeden koncový uzel už máme nakreslený. Znamená to, že ten druhý nemůže být žádný z dosud nakreslených; tím by se uzavřela kružnice. Na začátku jsme tedy měli jednu hranu s jejími dvěma koncovými uzly a potom vždy přidáváme současně jednu hranu a jeden uzel. Platí tedy následující věta:

Budiž $T = (U, H)$ strom o n uzlech. Potom T má $n - 1$ hran.

Snadno lze ukázat, že $n - 1$ je také nejmenší počet hran souvislého grafu o n uzlech. Má-li tedy graf o n uzlech méně než $n - 1$ hran, pak je nesouvislý.

Nyní si zvolme uzel $u_0 \in U$ a postupujme z něho tak, že vždy jdeme podél hrany z jednoho jejího koncového uzlu do druhého. Poněvadž jde o konečný graf, nemůžeme takto pokračovat donekonečna. Poněvadž jde o strom, nemůžeme přijít podruhé do téhož uzlu. Postup tedy musí skončit v nějakém uzlu u_1 stupně 1. Když potom postup opakujeme tak, že vyjdeme z u_1 místo z u_0 , musíme skončit opět v nějakém uzlu u_2 stupně 1 a samozřejmě $u_2 \neq u_1$. Platí tedy:

Každý konečný strom mající alespoň jednu hranu, obsahuje alespoň dva uzly stupně 1.

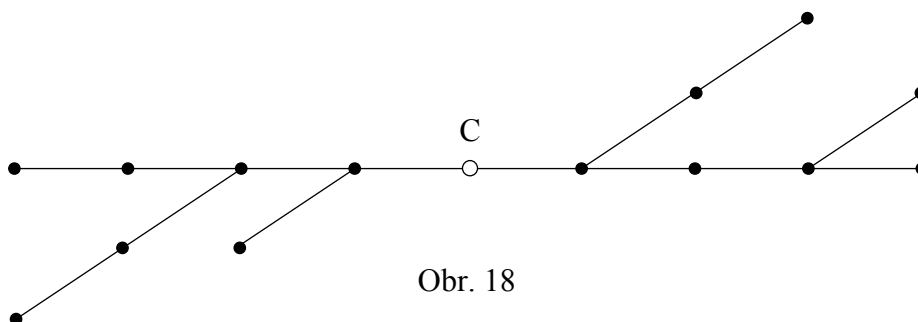
Z toho, že každá hrana stromu je acyklická, plyne toto tvrzení:

Nechť $T = (U, H)$ je strom o n uzlech, budiž $0 \leq k \leq n - 1$. Potom odstraněním libovolných k různých hran ze stromu T vznikne graf G o $k + 1$ komponentách, z nichž každá je strom.

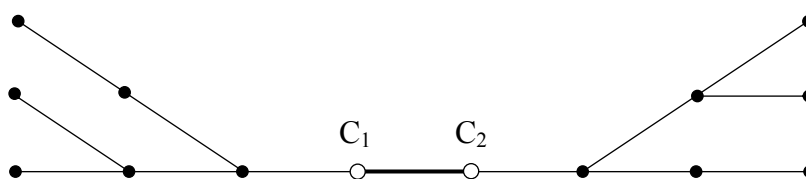
Graf, jehož všechny komponenty jsou stromy, se často nazývá les.

Významné jsou vlastnosti stromů týkající se vzdáleností mezi uzly. Platí:

- Ke každým dvěma uzlům u, v stromu $T = (U, H)$ existuje právě jedna cesta $P(u, v)$ z u do v . Lze tedy říci, že vzdálenost $d_G(u, v)$ je délka cesty z u do v .
- Budiž $T = (U, H)$ strom. Jestliže průměr $d(T)$ je sudé číslo, pak $d(T) = 2 \cdot r(T)$ a centrum stromu T se skládá z jediného uzlu. Jestliže $d(T)$ je liché číslo, pak $d(T) = 2 \cdot r(T) - 1$ a centrum stromu T se skládá ze dvou uzlů spojených hranou zvanou centrální hrana (viz obr. 18, 19).



Obr. 18



Obr. 19

Je-li $d_T(u, v) = d(T)$ pro některé uzly u, v stromu T , pak cesta $P(u, v)$ z u do v se nazývá diametrální cesta stromu T . Pak každá diametrální cesta stromu T obsahuje celé centrum stromu T .

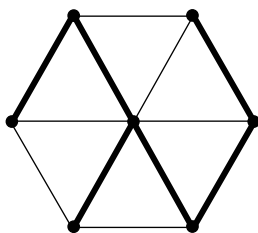
S pojmem stromu souvisí pojem kostry grafu.

Nechť $G = (U, H)$ je souvislý graf. Faktor grafu G , který je stromem, se nazývá kostra grafu G .

Kostra existuje v každém souvislém grafu. Je-li navíc graf sám stromem, má kostru právě jednu, a to sebe sama (jinak má nejméně tři kostry - dokažte).

Popišme si způsob, jak v daném souvislém grafu $G = (U, H)$ najdeme nějakou jeho kostru. Je-li graf G stromem, pak je svou vlastní kostrou a jsme hotovi. V opačném případě G obsahuje nějakou kružnici jako svůj podgraf. Všechny hrany této kružnice jsou cyklické. Zvolíme libovolnou z nich a odstraníme ji. Graf G_1 , který takto vznikne z G , je souvislý a je faktorem grafu G . Je-li to strom, jsme hotovi, jinak opakujeme postup s grafem G_1 místo G . Tento postup opakujeme, dokud nedojdeme ke grafu, jehož počet hran je o jednu menší než počet uzlů.

Na následujícím obr. 20 je graf s tučně vyznačenou kostrou.

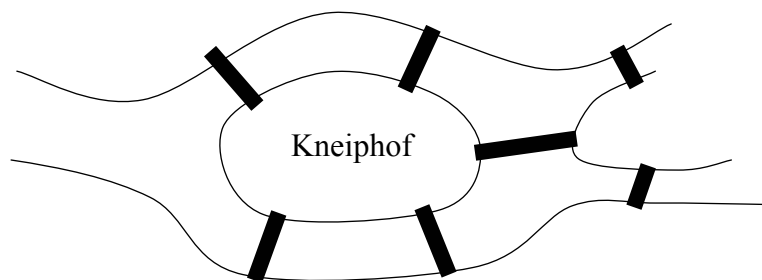


Obr. 20

Eulerovské tahy v grafu

Ve výkladu o souvislosti grafu jsme si definovali tah v grafu (otevřený nebo uzavřený). Uvažujme nyní o kreslení nákresu grafu. Máme-li v grafu G tah, pak všechny uzly i hrany tohoto tahu můžeme nakreslit skutečně jedním tahem bez zvednutí tužky a papíru. Existuje-li tah, který obsahuje všechny hrany souvislého grafu G , pak v takovém případě nakreslíme jedním tahem celý graf G .

A toho se týkala snad nejstarší věta z těch, které dnes zařazujeme do teorie grafů. Dnešní ruské město Kaliningrad bylo původně součástí Východního Pruska a jmenovala se Königsberg (česky bylo označováno jako Královec). Pojmenování prý vzniklo podle toho, že město založil český král Přemysl Otakar II. Městem protéká řeka Pregola (německy Pregel) a jsou v ní ostrovy, z nich jeden se nazýval Kneiphof. V 18. století stálo v Královci sedm mostů, jak je vidět na následujícím obr. 21.



Obr. 21

Mezi obyvateli města tehdy kolovala hádanka, zda lze projít městem tak, aby člověk přešel každý most právě jednou a nakonec se vrátil tam, odkud vyšel?

Tato hádanka zaujala švýcarského matematika Leonharda Eulera (1707 - 1783), který působil na universitě v nedalekém Petrohradě. Nejenže odpověděl na otázku, ale vytvořil i příslušnou teorii, kterou si stručně popíšeme.

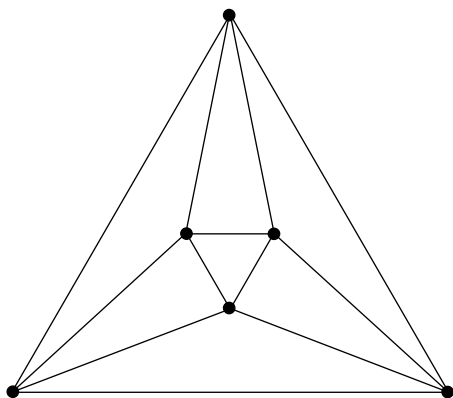
Mějme souvislý graf $G = (U, H)$. Tah v grafu G (otevřený nebo uzavřený) se nazývá eulerovský, jestliže se v něm každá hrana grafu G vyskytuje právě jednou.

Vidíme, že právě tehdy, když v grafu G existuje uzavřený eulerovský tah, můžeme náčrt grafu G nakreslit jedním tahem, přičemž vyjdeme z libovolného uzlu a v něm také skončíme (graf s uvedeným eulerovským tahem se nazývá eulerovský graf). Právě tehdy, když v grafu G existuje otevřený eulerovský tah z uzlu u do uzlu v , můžeme to provést tak, že začneme v uzlu u a skončíme v uzlu v nebo obráceně. Euler ukázal, kdy tyto případy nastávají (samozřejmě se musíme omezit na souvislé grafy).

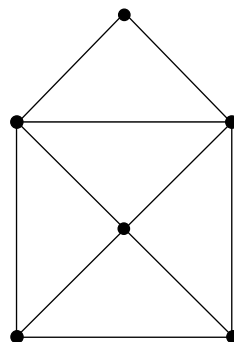
Platí:

V souvislém grafu G existuje uzavřený eulerovský tah právě tehdy, jestliže stupně všech uzlů jsou sudé. Otevřený eulerovský tah existuje právě tehdy, jestliže existují v G právě dva uzly lichého stupně (tah pak jde z jednoho do druhého).

Na obr. 22 vidíme graf s uzavřeným eulerovským tahem. Je to graf pravidelného osmistěnu (uzly odpovídají vrcholům, hrany hranám). Na obr. 23 je graf s otevřeným eulerovským tahem. (Nakreslete tyto obrázky jedním tahem.)



Obr. 22



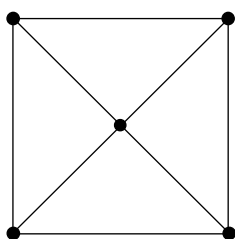
Obr. 23

Vzniká otázka, jak to bude při vyšším počtu uzlů lichého stupně? Připomeňme si, že tento počet musí být sudý, tj. bude to $2k$ pro celé číslo $k \geq 2$.

Platí:

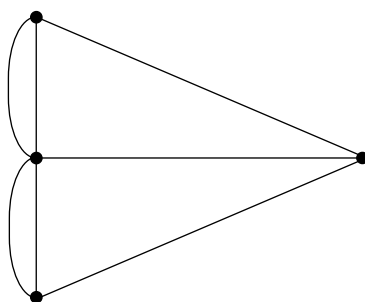
Nechť graf G má $2k$ uzlů lichého stupně, kde $k \geq 2$. Potom v grafu G existuje k otevřených tahů takových, že každá hrana grafu G je obsažena v jednom z nich.

Je zřejmé, že takový graf lze nakreslit k tahy. Pro $k = 2$ je ukázka takového grafu na následujícím obr. 24.



Obr. 24

Nakonec se opět vrátíme k hádance o mostech. Oba břehy i oba ostrovy znázorníme uzly. Dva uzly budou spojeny tolika hranami, kolik je mezi nimi mostů. (Nejde tedy přesně o graf podle naší definice, ale tvrzení pro něj platí.) Graf vidíme na obr. 25. Má čtyři uzly lichého stupně, tedy není eulerovský a žádanou procházku městem provést nelze.



Obr. 25

Nezávislé a dominantní množiny v grafu

Všimněme si podmnožin množiny uzlů grafu, které mají určité specifické vlastnosti.

Budiž $G = (U, H)$ graf, $J \subseteq U$ a necht' žádné dva uzly množiny J nejsou v G spojeny hranou. Potom J se nazývá nezávislá množina v grafu G .

Budiž $G = (U, H)$ graf, budiž $D \subseteq U$. Necht' množina D je taková, že každý uzel grafu G buď je v D , nebo je spojen hranou s uzlem z D . Potom D se nazývá dominantní množina v grafu G .

Obě vlastnosti jsou v teorii grafů i v jejich aplikacích velmi důležité. Platí:

- Je-li nějaká množina $J \subseteq U$ nezávislá a $J_0 \subseteq J$, potom je nezávislá i množina J_0 .
- Množina složená z jednoho uzlu grafu G je vždy nezávislá.
- Je-li nějaká množina $D \subseteq U$ dominantní v G a $D \subseteq D_0 \subseteq U$, potom je dominantní i množina D_0 .
- Množina U je vždy dominantní.

Maximální počet uzlů nezávislé množiny v konečném grafu G se nazývá číslo nezávislosti grafu G a značí se $\beta(G)$.

Platí:

- Pro kružnici C_n , n sudé je $\beta(C_n) = \frac{n}{2}$, pro liché n je $\beta(C_n) = \frac{n-1}{2}$.
- Pro úplný graf K_n je $\beta(K_n) = 1$.
- Pro úplný sudý graf $K_{m,n}$ je $\beta(K_{m,n}) = \max(m, n)$.

Minimální počet uzlů dominantní množiny v konečném grafu G se nazývá číslo dominance grafu G a značí se $\gamma(G)$.

Platí:

- Pro n dělitelné třemi je $\gamma(C_n) = \frac{n}{2}$, pro ostatní n je $\gamma(C_n) = \left\lfloor \frac{2n}{3} \right\rfloor + 1$.
- Pro úplný graf je $\gamma(K_n) = 1$.
- Pro úplný sudý graf $K_{m,n}$, kde $m \geq 2$, $n \geq 2$, je $\gamma(K_n) = 2$.

Jsou i další podobná čísla, která se týkají rozkladů. Necht' \mathfrak{Z} je rozklad množiny U takový, že každá jeho třída je nezávislá množina v G . Potom \mathfrak{Z} se nazývá uzlové obarvení grafu G . Minimální počet tříd uzlového obarvení grafu G se nazývá chromatické číslo neboli uzlová barevnost grafu G a značí se $\chi(G)$.

Mluvit o barvách v matematice je dost zvláštní. Má to však svůj historický důvod, který poznáme v příštím odstavci o rovinných grafech.

Často se chromatické číslo grafu G definuje přímo jako minimální počet barev, kterými lze obarvit uzly grafu G tak, aby uzly spojené hranou měly vždy různou barvu.

Platí:

- Pro kružnice C_n , n sudé, je $\chi(C_n) = 2$, pro n liché je $\chi(C_n) = 3$.
- Pro úplný graf je $\chi(K_n) = n$.
- Graf G alespoň s jednou hranou je sudý právě tehdy, je-li $\chi(G) = 2$.

Slovo „chromatický“ je z řeckého slova „chróma“, což znamená barva. Obdobou chromatického čísla je domatické číslo.

Necht' $<$ je rozklad množiny U takový, že každá jeho třída je dominantní množina v grafu G . Potom $<$ se nazývá domatický rozklad grafu G . Maximální počet tříd domatického rozkladu grafu G se nazývá domatické číslo grafu G a značí se $d(G)$. Slovo „domatický“ vzniklo ze slov „dominantní“ a „chromatický“ podobným způsobem jako slovo „smog“ ze slov „smoke“ a „fog“. Domatické číslo se totiž týká dominance a je určitou analogií chromatického čísla.

Platí:

- Pro kružnice C_n , kde n je dělitelné třemi, je $d(C_n) = 3$, pro jiná n je $d(C_n) = 2$.
- Pro úplný graf je $d(K_n) = n$.
- Pro úplný sudý graf $K_{m,n}$, kde $m \geq 2$, $n \geq 2$, jde $d(K_{m,n}) = 2$.
- Důležitá nerovnost je $d(G) \leq \delta(G) + 1$, kde $\delta(G)$ je minimální stupeň uzlu v grafu G .

Podobných čísel, která určitým způsobem charakterizují vlastnosti grafu, je v teorii grafů mnoho. Říká se jim číselné charakteristiky nebo číselné invarianty grafu.

Rovinné grafy

Podobně jako odstavec o eulerovských grafech, začneme i tento odstavec hádankou. Ve třech domech bydlí tři sousedé. Poblíž jejich domů jsou tři studně. Každý z obyvatel domů chce mít od svého domu cesty ke všem třem studnám. Nikdo však nechce, aby se některá jeho cesta křížovala se sousedovou. Je možné to zařídit?

Mějme graf G , jehož uzly jsou zmíněné tři domy a tři studně a jehož hrany představují popsání cesty. Je to zřejmě úplný sudý graf $K_{3,3}$. Pokud by bylo možné cesty postavit žádaným způsobem, znamenalo by to, že graf $K_{3,3}$ je rovinným grafem.

Víme, že graf lze znázornit nákresem, v němž uzly jsou znázorněny jako body a hrany jako úsečky nebo křivky. Jestliže lze všechny tyto body a úsečky či křivky umístit v rovině tak, aby zmíněné úsečky či křivky neměly nikde společný vnitřní bod (tedy aby se neprotínaly), příslušný graf G se nazývá rovinný.

Snadno bychom zjistili, že graf $K_{3,3}$ rovinný není, a tudíž odpověď na hádanku je záporná. Stejně tak není rovinný ani úplný graf K_5 o pěti uzlech. (Tento graf byl kdysi nazýván „muří noha“ a byl pokládán za magické znamení chránící proti nečistým silám. Je o tom zmínka i v Goethově „Faustovi“.)

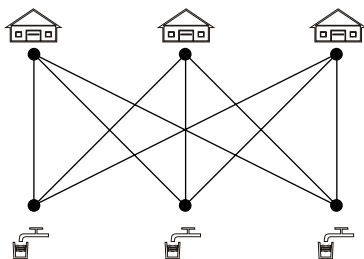
Není-li nějaký graf G rovinný, potom zřejmě není rovinný ani žádný graf, který z něho vznikne nahrazením některých jeho hran cestami délky alespoň dvě. Říkáme, že takový graf vznikl z grafu G dělením hran.

Rovněž je zřejmé, že pokud graf G obsahuje podgraf, který není rovinný, pak ani sám graf G není rovinný.

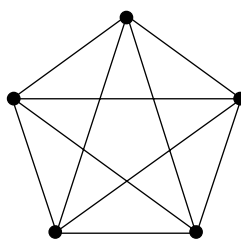
Vyjmenovali jsme si různé případy, kdy graf není rovinný. Platí (ovšem není vůbec lehké to dokázat), že ve všech ostatních případech graf rovinný je. Mluví o tom následující věta Kuratowského:

Konečný graf je rovinný právě tehdy, jestliže neobsahuje jako svůj podgraf graf $K_{3,3}$, graf K_5 ani žádný graf vzniklý z některého z těchto grafů dělením hran.

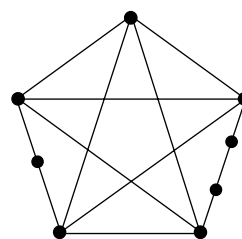
Na obr. 26 vidíme graf $K_{3,3}$, na obr. 27 graf K_5 a na obr. 28 graf vzniklý z grafu K_5 dělením hran.



Obr. 26



Obr. 27



Obr. 28

V souvislosti s rovinnými grafy se zmíníme ještě o tzv. problému čtyř barev.

Na politických mapách bývá vždy území každého státu obarveno nějakou barvou, přičemž sousední státy mají barvy různé. Pod názvem „problém čtyř barev“ se skrývala otázka, zda k obarvení libovolné takové mapy (splňující ovšem určité předpoklady) stačí čtyři barvy. Těm „územím států“ říkáme oblasti a předpokládáme, že každá oblast je ohraničena jednoduchou uzavřenou křivkou. Dvě oblasti pokládáme za sousední, pouze pokud společná část jejich hranic je oblouk křivky. (Pokud je tvořena pouze oddělenými izolovanými body, oblasti se za sousední nepokládají.) Mapě přiřadíme graf tím způsobem, že uvnitř každé oblasti zvolíme bod, který bude představovat uzel grafu. Dva uzly spojíme hranou právě tehdy, jestliže odpovídající oblasti spolu sousedí. Vzniklý graf je ovšem rovinný. Místo barvení oblastí pak barvíme uzly. Problém čtyř barev je tak převeden na otázku, zda platí $\chi(G) \leq 4$ pro libovolný rovinný graf G . A to je také historický důvod, proč se v teorii grafů barví, zatímco v jiných matematických oborech se mluví o rozkladech množin.

Autorem problému byl Francis Guthrie z Kapského Města roku 1852. Jeho řešiteli byli Kenneth Appel a Wolfgang Haken (někdy se uvádí i třetí jméno – Koch) z University státu Illinois v Urbaně roku 1976. Ti dokázali, že skutečně $\chi(G) \leq 4$ pro každý rovinný graf G . Takto tedy „problém čtyř barev“ vystřídala „věta o čtyřech barvách“.

Důkaz je důležitý vzhledem k minulosti i k budoucnosti. K minulosti, protože jde o řešení starého problému, který odolával dlouho přes sto let a k budoucnosti proto, že jde o jeden z prvních důkazů ryze teoretické věty provedených s podstatnou pomocí počítačů (dá se čekat, že postupně bude takových důkazů více).

Dva příklady aplikací teorie grafů

Ukážeme si dva příklady aplikací teorie grafů. V obou případech použijeme hranově ohodnoceného grafu.

Mějme graf G a každé jeho hraně h přiřadíme číslo z určité množiny čísel. Toto číslo pak nazveme ohodnocení hrany h . Jde tu v podstatě o zobrazení množiny H do určité množiny čísel (toto zobrazení se jmenuje hranové ohodnocení grafu G). V obou našich příkladech budeme uvažovat ohodnocení nezápornými reálnými čísly. (Poznamenejme, že existuje i uzlové ohodnocení grafu, které je definováno obdobně jako to hranové.)

Příklad 3.1 – Problém minimální kostry v grafu

Mějme několik míst, v nichž je třeba postavit telegrafní stanice. Tyto stanice mají být spojeny telegrafním vedením tak, aby bylo možno z kterékoliv z nich do kterékoliv jiné poslat telegram (buď přímo, nebo přes jiné stanice). Pro libovolné dvě projektované stanice jsou vypočteny finanční náklady potřebné na stavbu přímé linky mezi nimi. Sestrojte příslušnou telegrafní síť tak, aby celkové náklady byly minimální.

Sestrojíme ohodnocený graf G , jehož uzly budou projektované stanice. Dva uzly spojíme hranou, pokud má vůbec cenu stavět mezi nimi přímou linku. Každou hranu ohodnotíme číslem, které představuje výši nákladů na postavení přímé linky mezi jejími koncovými uzly. Nyní jde o to najít v grafu G podgraf G_0 , který by představoval graf hledané telefonní sítě. Zřejmě to musí být souvislý graf (to plyne z požadavku, aby z kterékoli stanice do kterékoli jiné bylo možno poslat telegram), navíc součet ohodnocení jeho hran (představuje celkové náklady na stavbu) musí být co nejmenší.

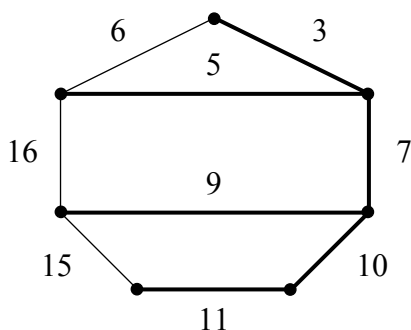
Aby součet ohodnocení byl nejmenší možný, je třeba, aby se v G_0 nevyskytovaly hrany, které by bylo možné považovat za zbytečné, zkrátka cyklické. Graf G_0 bude tedy strom a navíc faktor grafu G . Znamená to, že máme najít takovou kostru grafu G , která má ze všech koster grafu G nejmenší součet ohodnocení všech svých hran.

Existuje na to poměrně jednoduchý algoritmus. Sestavíme všechny hrany grafu G do konečné posloupnosti h_1, h_2, \dots, h_m tak, aby ohodnocení libovolné hrany bylo větší nebo rovno ohodnocením hran předcházejících, tj. $h_1 \leq h_2 \leq \dots \leq h_m$. Nakreslíme si nejdříve jednotlivé uzly grafu G . Pak postupně tvoříme podgrafy G_1, G_2, \dots, G_m grafu G . Graf G_1 má jedinou hranu h_1 . Jestliže máme sestrojen graf G_i pro nějaké přirozené číslo i , pak vezmeme

hranu h_{i+1} a zjistíme, zda jejím přidáním ke grafu G_i nevznikne graf obsahující kružnici. Jestliže ne, přidáme tuto hranu ke grafu G_i a vzniklý graf označíme G_{i+1} . V opačném případě položíme $G_{i+1} = G_i$. Jakmile takovýmto způsobem vyčerpáme celou naši posloupnost hran, dostaneme hledanou kostru.

Problém minimální kostry se někdy označuje jako Borůvkův problém podle českého matematika Otokara Borůvky (zabýval se jím i další český matematik Vojtěch Jarník).

Na následujícím obr. 29 vidíme ohodnocený graf G , v němž jsou tučně vyznačeny hrany jeho minimální kostry.



Obr. 29

Vidíme, že řešení není příliš složité. Podstatně odlišná situace by však nastala, pokud bychom požadovali, aby existovalo spojení mezi libovolnými dvěma stanicemi nejen při neporušené síti, ale i v případě, kdy se libovolná jedna linka přeruší. Zde se nepodařilo najít dostatečně efektivní postup, který by při libovolném výchozím grafu dal požadované řešení.

Příklad 3.2 – Čínský problém listonoše.

Listonoš musí denně projít všechny ulice svého obvodu a vrátit se na místo, odkud vyšel. Jde mu o to, aby tato cesta byla co nejkratší.

V anglicky psané literatuře se toto označuje jako „Chinese Postman Problem“ a do češtiny se to často chybně překládá jako „problém čínského listonoše“. Ve skutečnosti jde o „čínský problém listonoše“. Jde o problém listonoše bez ohledu na jeho národnost. A je to problém čínský, protože jeho autorem je čínský matematik (jeho jméno se někdy uvádí jako Kwan (anglickým pravopisem), jindy jako Guan (podle pravidel standardního přepisu čínštiny do latinky)).

Listonošův obvod lze znázornit ohodnoceným grafem G , jehož hrany jsou jednotlivé ulice (uzly jsou samozřejmě rozcestí), každá hrana je ohodnocena délkou příslušné ulice. Měli bychom najít spojení z některého uzlu u_0 takové, aby se v něm každá hrana grafu G vyskytovala alespoň jednou a aby součet ohodnocení hran tohoto spojení byl ze všech spojení splňujících dané podmínky minimální.

Už víme, co je eulerovský tah a eulerovský graf. Listonoš to má nejjednodušší tehdy, je-li graf G eulerovský. Pak může procházet svým obvodem tak, že jde po uzavřeném eulerovském tahu. Každou ulici projde právě jednou a vrátí se nakonec na to místo, odkud vyšel. Poněvadž každou ulici musí projít aspoň jednou, je tato jeho cesta zřejmě ze všech možných cest nejlepší (žádnou ulicí neprochází vícekrát).

Horší je to v případě, kdy v grafu G existují uzly lichého stupně. Ukažme si řešení v tomto případě.

Nechť U_0 je množina všech uzlů lichého stupně v grafu G . Sestrojme nový ohodnocený graf G_0 . Jeho množinou uzlů bude U_0 a každé dva různé uzly v něm budou spojeny hranou (bude to tedy úplný graf). Pro každé dva uzly u, v množiny U_0 najdeme v grafu G takovou cestu z u do v , aby součet ohodnocení jejích hran byl minimální možný. Tento součet ohodnocení označíme $d_0(u, v)$. V grafu G_0 nyní každou hranu ohodnotíme číslem $d_0(u, v)$, kde u, v jsou uzly spojené touto hranou.

Budiž n počet uzlů grafu G_0 (víme, že je to sudé číslo). V G_0 existují množiny složené z $\frac{n}{2}$ hran, z nichž žádné dvě nejsou incidentní s tímž uzlem (takovým množinám říkáme párování grafu G_0). Ze všech párování vyberme to, u něhož je součet ohodnocení minimální (existuje na to algoritmus, který zde popisovat nebudeme). Hrany tohoto párování i s příslušným ohodnocením přidáme ke grafu G (vzniknou tak jakési fiktivní ulice). V takto získaném grafu G_1 není vyloučeno, že některé dva uzly jsou spolu spojeny dvěma hranami. To však zde nevádí, neboť podstatné je to, že v G_1 má každý uzel sudý stupeň (k uzlům, které měly sudý stupeň v G , nepřibyla žádná hrana a k těm, které měly lichý stupeň přibýlo po jedné). Graf G_1 je tedy eulerovský. Můžeme jej kreslit jedním tahem a toto kreslení určuje listonošovu cestu. Kreslíme-li hranu grafu G , odpovídá to tomu, že listonoš má jít po odpovídající ulici. Kreslíme-li hranu grafu G_0 z uzlu u do uzlu v , odpovídá to tomu, že listonoš jde z rozcestí u na rozcestí v nejkratší možnou cestou, tedy cestou délky $d_0(u, v)$.

Takto je tedy rozřešen listonošův problém. Nepopisovali jsme zde ovšem ani algoritmy pro nalezení cesty mezi dvěma uzly s minimálním součtem ohodnocení hran, ani algoritmy pro nalezení párování s minimálním součtem ohodnocení hran (to by bylo příliš složité).

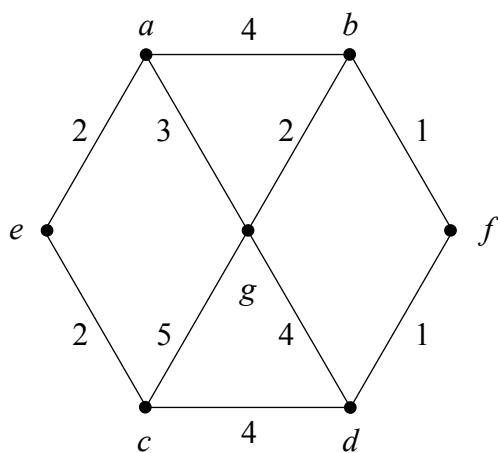
Ukažme si příklad. Zkoumáme graf G na obr. 30. Tento graf obsahuje čtyři uzly lichého stupně, a to a, b, c, d . Měli bychom sestavit graf G_0 . Nejkratší cesta z a do b v G obsahuje jedinou hranu s ohodnocením 4 a podobně je tomu u uzlů c a d .

Tedy $d_0(a, b) = d_0(c, d) = 4$. Nejkratší cesta z a do c obsahuje uzel e a má součet ohodnocení hran 4; tedy $d_0(a, c) = 4$. Nejkratší cesta z b do d obsahuje uzel f a $d_0(b, d) = 2$. Nejkratší cesta z a do d obsahuje uzly b a f a $d_0(a, d) = 6$. Konečně nejkratší cesta z b do c obsahuje uzly f a d a $d_0(b, c) = 6$. Graf G_0 vidíme na obr. 31. Je to úplný graf o čtyřech uzlech a snadno bychom zjistili, že v něm existují právě tři párování. Jedno z nich se skládá z hran $\{a, b\}, \{c, d\}$, druhé z hran $\{a, c\}, \{b, d\}$, třetí z hran $\{a, d\}, \{b, c\}$. U všech určíme součet ohodnocení hran. U prvního je to 8, u druhého 6 a u třetího 14. Párování s nejmenším součtem ohodnocení je to, které se skládá z hran $\{a, c\}, \{b, d\}$. Přidáme tedy hrany $\{a, c\}$ a $\{b, d\}$ s příslušným ohodnocením ke grafu G a dostaneme graf G_1 na obr. 32.

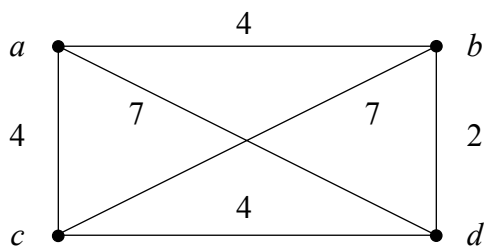
Jsou různé možnosti, jak projít graf G_1 jedním tahem. Jednu z těchto možností bychom mohli zapsat jako $a - b - f - d - b - g - d - c - g - a - c - e - a$. Listonoš tedy bude procházet uzly grafu G v uvedeném pořadí. Pokud mezi uzly, které za sebou bezprostředně následují, existuje ulice, půjde listonoš po ní. Pouze z d do b a z a do c musí jít jiným způsobem. Z d do b půjde přes f , z a do c půjde přes e . Tedy pouze hrany (čili ulice) $\{a, e\}, \{b, f\}, \{c, e\}, \{d, f\}$ projde listonoš dvakrát, všechny ostatní jednou.

Je zřejmé, že tento problém se netýká pouze listonošů, ale že jde o významnou otázku plánování dopravy.

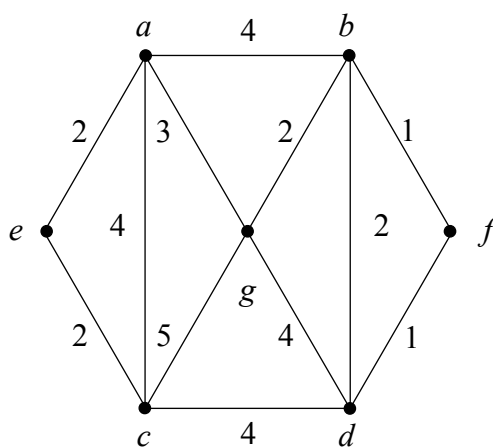
Obdobný je problém obchodního cestujícího, pro jehož řešení neexistuje univerzálně použitelný algoritmus. Obchodní cestující má vykonat cestu, při níž navštíví alespoň jednou každé město z dané množiny měst. Vzdálenost mezi každými dvěma z těchto měst je známa. Zase jde o otázku, jak tuto cestu podniknout, aby se ujelo co nejméně kilometrů. I zde se ovšem používá ohodnocených grafů.



Obr. 30



Obr. 31



Obr. 32

Orientované grafy

Zabývali jsme se zde neorientovanými grafy. Kromě nich existují také grafy, kterým se říká orientované. Liší se od neorientovaných tím, že jejich hrany jsou uspořádané dvojice uzlů.

Orientovaný graf je uspořádaná dvojice $G = (U, H)$, kde U je množina prvků zvaných uzly a H je podmnožina množiny všech uspořádaných dvojic prvků množiny U (tedy podmnožina kartézského součinu $U \times U$). Prvky množiny H se nazývají orientované hrany grafu.

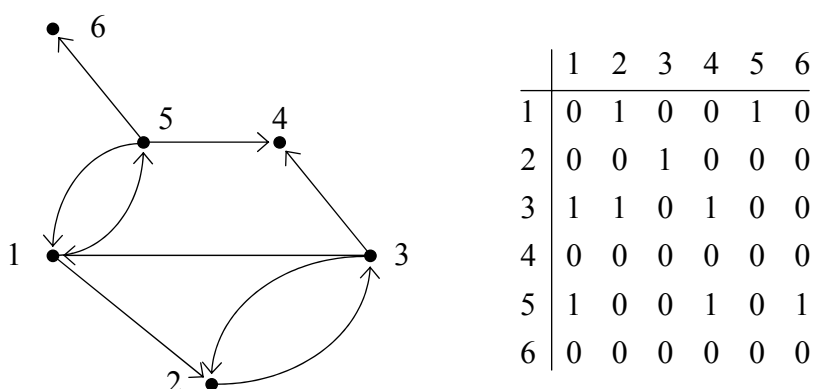
Rovněž orientované grafy znázorňujeme nákresem. Hrana (u, v) se znázorní stejně jako hrana $\{u, v\}$ v neorientovaném grafu a přikreslí se k ní šipka od u do v . Připouští se možnost, že mezi uzly u, v existují dvě hrany $(u, v), (v, u)$.

Je-li $h = (u, v)$, pak říkáme, že u je počáteční a v koncový uzel hrany h a že hrana h jde z uzlu u do uzlu v .

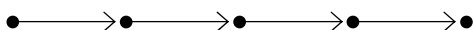
Orientovaný graf lze také určit jeho maticí sousednosti, kde v i -tém řádku a j -tém sloupci je jednička, pokud existuje hrana z uzlu i do uzlu j , jinak je tam nula (obr. 33).

Analogií cesty v orientovaném grafu je dráha (obr. 34), analogií kružnice je cyklus (obr. 35). Analogií souvislosti je silná souvislost. Orientovaný graf $G = (U, H)$ je silně souvislý, jestliže pro libovolné dva jeho uzly u, v existuje dráha z u do v a dráha z v do u .

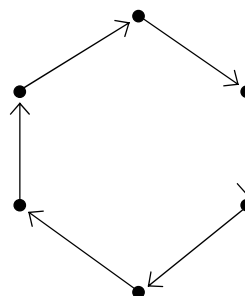
I orientované grafy mají četné aplikace, například znázornění uliční sítě s jednosměrnými ulicemi, rodokmeny či vývojové diagramy při programování počítačů.



Obr. 33



Obr. 34



Obr. 35

Přílohy

Použité značení

Logika

| | |
|------------------------|---|
| p, q | ... výroky, |
| $\neg p, \overline{p}$ | ... negace, |
| $p \wedge q$ | ... logická spojka „a“ (konjunkce), |
| $p \vee q$ | ... logická spojka „nebo“ (disjunkce), |
| $p \rightarrow q$ | ... implikace (jestliže), |
| $p \leftrightarrow q$ | ... ekvivalence (právě když), |
| $p \Leftrightarrow q$ | ... logická ekvivalence, |
| $P(x_1, \dots, x_n)$ | ... výroková funkce, |
| $\forall x P(x)$ | ... pro všechna x platí $P(x)$; obecný (velký) kvantifikátor, |
| $\exists x P(x)$ | ... existuje x , pro které platí $P(x)$; existenční kvantifikátor, |
| $\exists! x P(x)$ | ... existuje právě jedno x , pro které platí $P(x)$. |

Množiny

| | |
|-----------------------|---|
| $\emptyset, \{ \}$ | ... prázdná množina, |
| U | ... univerzum, |
| N | ... množina přirozených čísel (včetně nuly), |
| N^+ | ... množina kladných přirozených čísel, |
| Z | ... množina celých čísel, |
| Q | ... množina racionálních čísel, |
| R | ... množina reálných čísel, |
| C | ... množina komplexních čísel, |
| $\{a_1, \dots, a_n\}$ | ... množina skládající se z prvků a_1, \dots, a_n ; neuspořádaná n -tice, |
| $\{a \mid V(a)\}$ | ... množina prvků s vlastností V , |
| $x \in A$ | ... x je prvkem množiny A (bodová inkluze), |
| $x \notin A$ | ... x není prvkem množiny A , |
| $A \subseteq B$ | ... A je podmnožinou B , |
| $A \subset B$ | ... A je vlastní podmnožinou množiny B , |
| $A \cap B$ | ... průnik množin A, B , |
| $A \cup B$ | ... sjednocení množin A, B , |
| \overline{A} | ... doplněk množiny A , |
| $A - B$ | ... rozdíl množin $A - B$, |
| $A \div B$ | ... symetrický rozdíl (diference) množin A, B , |
| $P(A)$ | ... potenční množina, |
| $ A $ | ... počet prvků (mohutnost, kardinalita) množiny A . |

Relace a funkce

| | |
|-------------------------------|---|
| $A \times B$ | ... kartézský součin množin A, B , |
| (a, b) | ... uspořádaná dvojice prvků a, b , |
| $R \subseteq A \times B$ | ... R je relace z A do B , |
| $aRb, (a, b) \in R$ | ... a je v relaci s b , |
| $a \not R b, (a, b) \notin R$ | ... a není v relaci s b , |
| $[a]$ | ... třída ekvivalence určená prvkem a , |
| $a b$ | ... a dělí b (tj. beze zbytku), |
| $a \nmid b$ | ... a nedělí b , |
| $a \equiv b \pmod{m}$ | ... a je kongruentní s b modulo m , |
| $a \not\equiv b \pmod{m}$ | ... a není kongruentní s b modulo m , |
| $a = (b \bmod m)$ | ... a je zbytek při dělení b číslem m , |
| $f : A \rightarrow B$ | ... funkce A do B , |
| $f(a)$ | ... hodnota funkce f v bodě a , |
| $\{a_n\}_{n=0}^{\infty}$ | ... číselná posloupnost, |
| $\lceil x \rceil$ | ... horní celá část čísla x , |
| $\lfloor x \rfloor$ | ... dolní celá část čísla x , |
| $\{x\}$ | ... lomená část x , |
| $m!$ | ... m faktoriál, |
| $\ln x$ | ... přirozený logaritmus čísla x . |

Teorie grafů

| | |
|---|---|
| $G = (U, H)$ | ... (neorientovaný) graf s množinou uzlů U a hran H , |
| $U(G)$ | ... množina uzlů grafu G , |
| $H(G)$ | ... množina hran grafu G , |
| $\{u, v\}$ | ... (neorientovaná) hrana, |
| (u, v) | ... orientovaná hrana, |
| $\deg_G(v)$ | ... stupeň uzlu v , |
| K_n | ... úplný graf s n uzly, |
| $K_{n,m}$ | ... úplný bipartitní graf, |
| C_n | ... kružnice s n uzly, |
| $a = u_1, h_1, \dots, h_k, u_{k+1} = b$ | ... sled, tah, cesta z uzlu a do b , (vypsána jako posloupnost uzlů a hran), |
| $a = u_1, \dots, u_{k+1} = b$ | ... sled, tah, cesta z uzlu a do b , (vypsána jako posloupnost uzlů). |

Literatura

- J. Anděl: Matematika náhody. Matfyzpress, Praha 2000.
- R. A. Brualdi: Introductory Combinatorics. Prentice-Hall, 1997.
- F. Calda, V. Dupač: Matematika pro gymnázia. Kombinatorika, pravděpodobnost, statistika. Prometheus, Praha 1994.
- R. P. Grimaldi: Discrete and Combinatorial Mathematics. Addison-Wesley, 1998.
- D. E. Knuth: The Art of Computer Programming. Volume 1: Fundamental Algorithms, Addison-Wesley, 1997.
- D. E. Knuth: The Art of Computer Programming. Volume 2: Seminumerical Algorithms, Addison-Wesley, 1997.
- D. E. Knuth: The Art of Computer Programming. Volume 3: Sorting and Searching, Addison-Wesley, 1998.
- L. Kosmák: Množinová algebra. MU, Brno 1995.
- J. Matoušek, J. Nešetřil: Kapitoly z diskrétní matematiky. Karolinum, Praha 2000.
- I. Pohl, A. Shaw: The Nature of Computation: An Introduction to Computer Science. Computer Science Press, 1981.
- K. H. Rosen: Handbook of Discrete and Combinatorial Mathematics. CRC Press, 2000.
- K. H. Rosen: Discrete Mathematics and Its Applications. McGraw-Hill, 1999.
- N. R. Scott: Computer Number Systems and Arithmetic. Prentice Hall, 1985.
- J. Sedláček: Úvod do teorie grafů. Academia, Praha 1981.
- B. Zelinka: Rovinné grafy. Mladá fronta, Praha 1977.

Relevantní zdroje na internetu

- http://www-history.mcs.st-and.ac.uk/history/HistTopic/Beginnings_of_set_theory.html#61
- <http://www.mcs.rbjones.com/rbjpub/logic/loh025.htm>
- <http://www-formal.stanford.edu:80/clt/ARS/systems.html>
- <http://www-cad.eecs.berkeley.edu/~fmang/paradox.html>
- <http://www.maths.usyd.edu.au:800/u/magma/>
- <http://wwwmaths.anu.edu.au/research.groups/algebra/GAP/www/gap.html>
- http://msm.yahoo.com/Science/Mathematics/Graph_Theory/
- <http://wwwis.win.tue.nl/~percy/my/link-gd.html>
- <http://www.ing.unlp.edu.ar/cetad/mos/Hamilton.html>
- <http://www.math.gatech.edu/~thomas/FC>
- <http://www.cs.sunysb.edu/~algorithm/>