

Grafy

Hoci prvá práca z teórie grafov pochádza z roku 1736 a mnoho zaujímavých výsledkov v tejto oblasti bolo získaných v devätnástom storočí, prudký rozmach teórie grafov začína až po roku 1920. Nepochybne jedným z dôvodov veľkého záujmu o teóriu grafov je jej aplikovateľnosť v mnohých oblastiach, ako napr. informatika, chémia, elektrotechnika, lingvistika a ekonómia.

Orientované a neorientované grafy

Neorientovaný graf je usporiadaná trojica $G = (V, E, e)$, kde V, E sú konečné množiny, $V \neq \emptyset, V \cap E = \emptyset$ a e je zobrazenie $e : E \rightarrow W$, pričom W je množina všetkých jednoprvkových a dvojprvkových podmnožín množiny V .

Prvky množiny V sa nazývajú **vrcholy**, prvky množiny E sa nazývajú **hrany** (**neorientované**). Zobrazenie e sa nazýva **incidencia**. Ak h je hrana, a, b sú vrcholy a $e(h) = \{a, b\}$ (vrcholy a, b môžu byť aj totožné), tak a, b sa nazývajú **krajné vrcholy hrany** h . Hovoríme tiež, že hrana h **inciduje** (**je incidentná**) s vrcholmi a, b .

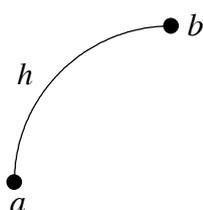
Príklad. Nech $G_1 = (V, E, e_1)$, kde $V = \{1, 2, 3, 4\}$, $E = \{h_1, h_2, h_3, h_4, h_5, h_6\}$,
 $e_1 : e_1(h_1) = \{1, 4\}$, $e_1(h_2) = \{2, 4\}$, $e_1(h_3) = \{2, 3\}$, $e_1(h_4) = e_1(h_5) = \{3, 4\}$, $e_1(h_6) = \{2\}$.
 G_1 je zrejme neorientovaný graf. ■

Orientovaný graf je usporiadaná trojica $G = (V, E, e)$, kde V, E sú konečné množiny, $V \neq \emptyset, V \cap E = \emptyset$ a e je zobrazenie $e : E \rightarrow V^2$. Prvky množiny V sa nazývajú **vrcholy**, prvky množiny E sa nazývajú **orientované hrany** a zobrazenie e sa nazýva **incidencia**. Ak a, b sú vrcholy, h je orientovaná hrana a $e(h) = (a, b)$, tak a, b sú **krajné vrcholy orientovanej hrany** h , špeciálne: vrchol a sa nazýva **počiatočný vrchol orientovanej hrany** h a vrchol b **koncový vrchol orientovanej hrany** h .

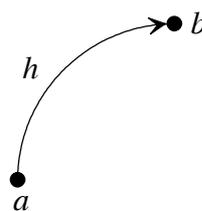
Príklad. Nech $G_2 = (V, E, e_2)$, kde $V = \{1, 2, 3, 4\}$, $E = \{h_1, h_2, h_3, h_4, h_5, h_6\}$,
 $e_2 : e_2(h_1) = (1, 4)$, $e_2(h_2) = (4, 2)$, $e_2(h_3) = (3, 2)$, $e_2(h_4) = e_2(h_5) = (3, 4)$, $e_2(h_6) = (2, 2)$. G_2
je orientovaný graf. ■

Dohoda. Aby sme zostručnili vyjadrovanie, budeme ďalej v názve „neorientované grafy“ slovo „neorientované“ vynechávať a písať len „grafy“. Ak budeme chcieť vyjadriť, že niečo sa týka neorientovaných aj orientovaných grafov, použijeme zápis: ...grafy [orientované grafy]... alebo ...[orientované] grafy... .

Grafy [orientované grafy] budeme znázorňovať aj graficky a to takýmto spôsobom: vrcholy [orientovaného] grafu G znázorníme ako body roviny, ktoré označíme rovnako ako samotné vrcholy. Nech h je [orientovaná] hrana s krajnými vrcholmi a, b . Potom hranu h (v prípade neorientovaného grafu) znázorníme ako čiaru spájajúcu body a, b (obr. 30). Orientovanú hranu h s počiatočným vrcholom a a koncovým b znázorníme ako čiaru spájajúcu body a, b so šípku pri koncovom vrchole b (obr. 31). Takto vytvorený objekt budeme nazývať **diagram** [orientovaného] grafu G .

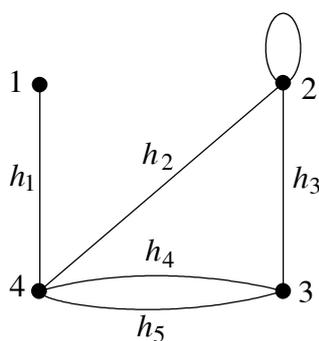
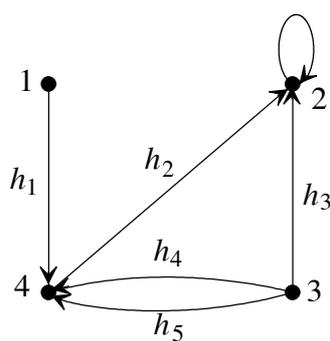


Obr. 30



Obr. 31

Príklad. Diagramy grafu G_1 a orientovaného grafu G_2 z predchádzajúcich príkladov sú na obr. 32 a 33. ■

Obr. 32. Diagram grafu G_1 Obr. 33. Diagram orientovaného grafu G_2

Ak hrana h [orientovaného] grafu G inciduje s dvoma rôznymi vrcholmi, nazýva sa [orientovaná] **linka** a ak inciduje len s jedným vrcholom, nazýva sa [orientovaná] **slučka**.

Príklad. V grafe G_1 (obr. 32) hrany h_1, h_2, h_3, h_4, h_5 sú linky a hrana h_6 je slučka. ■

Vrcholy u, v [orientovaného] grafu G sa nazývajú *susedné*, ak incidujú s tou istou hranou. Vrchol, ktorý neinciduje so žiadnou linkou, sa nazýva *izolovaný vrchol*.

Násobnosťou orientovanej hrany s počiatočným vrcholom u a koncovým vrcholom v v orientovanom grafe G nazývame počet orientovaných hrán, ktorých počiatočný vrchol je u a koncový je v . Toto číslo budeme označovať $m^+(u, v)$.

Výstupný stupeň vrcholu u orientovaného grafu G je počet orientovaných hrán s počiatočným vrcholom u . Toto číslo budeme označovať $st^+(u)$.

Vstupný stupeň vrcholu u je počet orientovaných hrán s koncovým vrcholom u . Toto číslo budeme označovať $st^-(u)$.

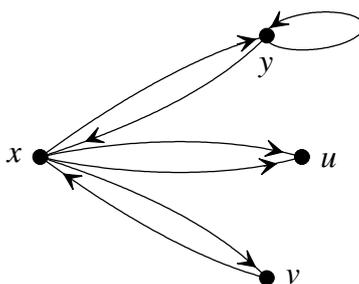
Násobnosť hrany s krajnými vrcholmi u, v v grafe G je počet hrán grafu G s krajnými vrcholmi u, v . Budeme ju označovať $m(u, v)$.

Stupeň vrcholu u v [orientovanom] grafe G je počet [orientovaných] hrán, ktoré incidujú s vrcholom u , pričom [orientovaná] slučka sa počíta dvakrát. Stupeň vrcholu u budeme označovať $st(u)$.

Príklad. V orientovanom grafe, ktorého diagram je na obr. 34, je

$$\begin{aligned} st^+(u) &= 0, st^-(u) = 2, st(u) = 2 \\ st^+(v) &= 1, st^-(v) = 1, st(v) = 2 \\ st^+(x) &= 3, st^-(x) = 2, st(x) = 5 \\ st^+(y) &= 2, st^-(y) = 2, st(y) = 4 \end{aligned}$$

$$\begin{aligned} m^+(u, v) &= 0, m^+(u, x) = 0, \\ m^+(v, x) &= 1, \\ m^+(x, u) &= 2, m^+(x, y) = 1, \\ m^+(y, x) &= 1, m^+(y, y) = 1. \end{aligned}$$



Obr. 34

Veta. Pre každý vrchol u orientovaného grafu G platí

$$st(u) = st^+(u) + st^-(u)$$

Dôkaz. Vyplýva priamo z definícií stupňa, vstupného a výstupného stupňa vrcholu. 📌

Veta. Súčet stupňov všetkých vrcholov [orientovaného] grafu je párny a rovná sa dvojnásobku počtu jeho [orientovaných] hrán.

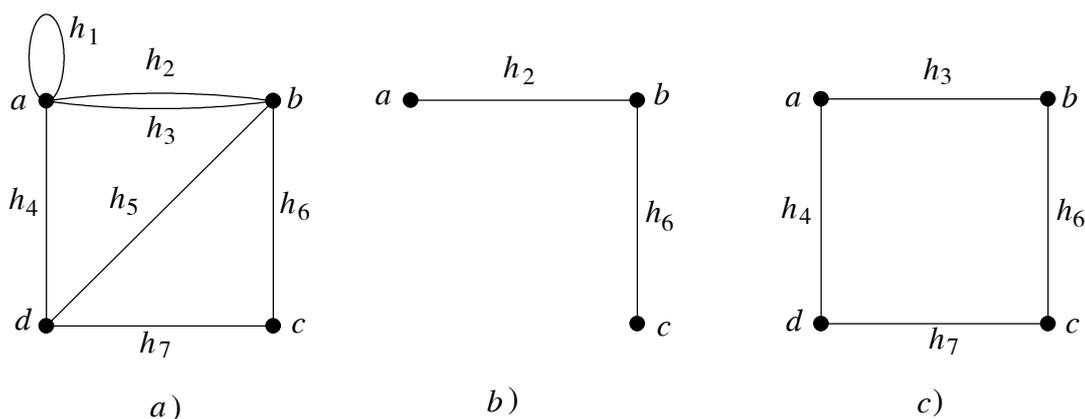
Dôkaz. Tvrdenie vety vyplýva z toho, že každá linka zvyšuje stupeň dvoch rôznych vrcholov o 1 a každá slučka zvyšuje stupeň jedného vrcholu o 2. 🔔

Veta. Počet všetkých vrcholov [orientovaného] grafu nepárneho stupňa je párny.

Dôkaz. Predpokladajme, že počet všetkých vrcholov nepárneho stupňa je nepárny. Potom súčet stupňov všetkých vrcholov nepárneho stupňa je nepárny. Z toho a z faktu, že súčet stupňov vrcholov párneho stupňa je párny, vyplýva, že súčet stupňov všetkých vrcholov je nepárny. To je však v spore s predchádzajúcou vetou. 🔔

[Orientovaný] graf $G' = (V', E', e')$ sa nazýva **podgraf** [orientovaného] grafu $G = (V, E, e)$, ak $V' \subset V$, $E' \subset E$, e' je zúžením incidencie e na množinu hrán E' .

Príklad. Na obr. 35b, 35c sú podgrafy grafu G , ktorého diagram je na obr. 35a. ■



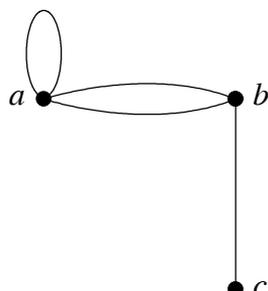
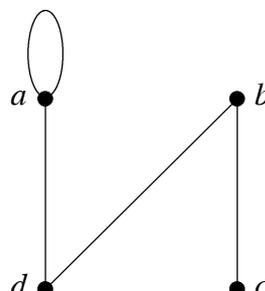
Obr. 35. a) Graf G b), c) Podgrafy grafu G

Nech $G = (V, E, e)$ je [orientovaný] graf, $V' \subset V$, $V' \neq \emptyset$. Podgraf $G(V')$ [orientovaného] grafu G , ktorého množina vrcholov je V' a množina hrán je určená vlastnosťou: ak $u, v \in V'$, tak $G(V')$ obsahuje všetky hrany s krajnými vrcholmi u, v [orientovaného] grafu G , sa nazýva **indukovaný podgraf**.

Nech $G = (V, E, e)$ je [orientovaný] graf, $E' \subset E$. Podgraf $G(E')$ [orientovaného] grafu G , ktorého množina vrcholov je V a množina hrán je E' , sa nazýva **faktor [orientovaného] grafu G** .

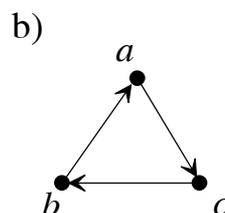
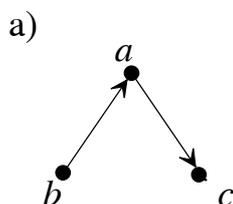
Príklad. Nech $V' = \{a, b, c\}$, $E' = \{h_1, h_4, h_5, h_6\}$, G je graf na obr. 35a. Indukovaný podgraf $G(V')$ grafu G je na obr. 36. Na obr. 38 je faktor $G(E')$ grafu G . ■

[Orientovaný] graf sa nazýva **jednoduchý**, ak násobnosť každej jeho hrany je najviac 1. [Orientovaný] graf, ktorý nie je jednoduchý sa nazýva [**orientovaný**] **multigraf**. [Orientovaný] graf, ktorý neobsahuje žiadne slučky, sa nazýva **bezslučkový**.

Obr. 36. Indukovaný podgraf grafu G Obr. 37. Faktor grafu G

Cvičenia

- Nakreslite diagram grafu, ktorý má požadované vlastnosti alebo vysvetlite, prečo taký graf neexistuje:
 - šesť vrcholov tretieho stupňa;
 - päť vrcholov tretieho stupňa;
 - šesť vrcholov, štyri hrany;
 - štyri vrcholy, ktorých stupne sú 1, 2, 3, 4;
 - štyri hrany, štyri vrcholy, ktorých stupne sú 1, 2, 3, 4;
 - jednoduchý bezslučkový graf, šesť vrcholov, ktorých stupne sú 1, 2, 3, 4, 5, 5.
- Je možné strany a uhlopriečky pravidelného deväťuholníka zafarbiť modrou a červenou farbou (každú úsečku len jednou farbou) tak, aby z každého vrcholu vychádzali práve 3 červené úsečky?
- Nájdite všetky podgrafy daného grafu:



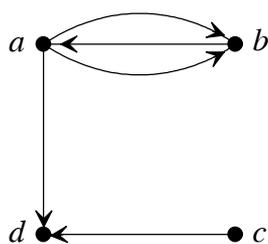
Reprezentácia [orientovaných] grafov

Doteraz sme reprezentovali (zadávali) graf obvykle grafickou formou - jeho diagramom. Na analýzu grafov pomocou výpočtovej techniky sú vhodnejšie iné reprezentácie grafov.

Uvedieme štyri najzákladnejšie spôsoby reprezentácie grafov.

Prvý spôsob vychádza priamo z definície grafu. [Orientovaný] graf budeme reprezentovať pomocou *zoznamu vrcholov a hrán*. Do tohto zoznamu je potrebné vložiť aj informáciu o incidencii hrán a vrcholov, čo môžeme urobiť tak, že hranu h budeme reprezentovať trojicou (h, u, v) , kde u, v sú krajné vrcholy hrany h , v prípade orientovaného grafu je u počiatkový a v koncový vrchol hrany h .

Príklad. Diagram orientovaného grafu zadaného zoznamom vrcholov $V = \{a, b, c, d\}$ a hrán $E = \{(h_1, a, b), (h_2, a, b), (h_3, b, a), (h_4, a, d), (h_5, c, d)\}$ je na obr. 38. ■



Obr. 38

Druhým spôsobom reprezentácie, avšak len jednoduchých [orientovaných] grafov, je *systém okolí vrcholov*.

Okolie vrcholu u grafu $G = (V, E, e)$ je množina

$$V_u = \{x \in V; u, x \text{ sú susedné vrcholy}\}.$$

Okolie vrcholu u orientovaného grafu $G = (V, E, e)$ je množina

$$V_u = \{x \in V; u \text{ je počiatkový a } x \text{ koncový vrchol tej istej hrany}\}.$$

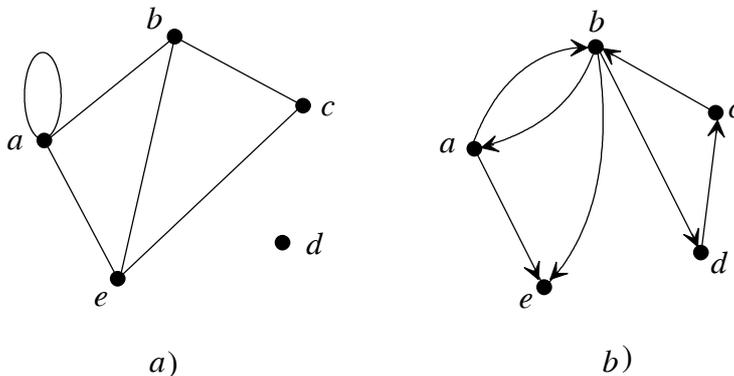
Jednoduchý [orientovaný] graf G môžeme reprezentovať systémom okolí jeho vrcholov, t.j. množinou $W = \{V_u; u \in V\}$.

Príklad. Daný je a) graf, b) orientovaný graf, systémom okolí

a) $W = \{V_a = \{a, b, e\}, V_b = \{a, c, e\}, V_c = \{b\}, V_d = \emptyset, V_e = \{a, b, c\}\}$

b) $W = \{V_a = \{b, e\}, V_b = \{a, d, e\}, V_c = \{b\}, V_d = \{c\}, V_e = \emptyset\}$

Ich diagramy sú na obr. 39. ■



Obr. 39

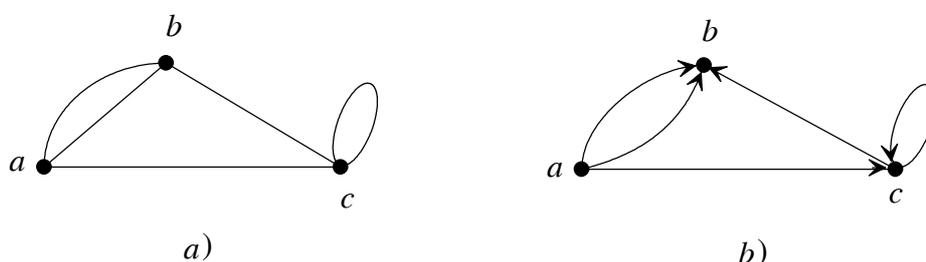
K ďalším dvom reprezentáciám grafov sa využívajú matice.

Nech G je [orientovaný] graf s n vrcholmi. **Maticou susednosti** [orientovaného] grafu G pri usporiadaní jeho vrcholov v_1, v_2, \dots, v_n sa nazýva matica

$$(m_{ij})_{i,j \in \{1, \dots, n\}},$$

kde

$$m_{ij} = \begin{cases} m^+(v_i, v_j) & \text{, pre orientovaný graf} \\ m(v_i, v_j) & \text{, pre neorientovaný graf} \end{cases}.$$



Obr. 40

Príklad.

a) Maticou susednosti grafu z obr. 40a pri usporiadaní vrcholov a, b, c je

$$\begin{pmatrix} 0 & 2 & 1 \\ 2 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}.$$

Ak zvolíme iné usporiadanie vrcholov, napr. b, c, a , tak maticou susednosti pri tomto usporiadaní je

$$\begin{pmatrix} 0 & 1 & 2 \\ 1 & 1 & 1 \\ 2 & 1 & 0 \end{pmatrix}.$$

b) Maticou susednosti orientovaného grafu z obr. 40b pri usporiadaní vrcholov a, b, c je

$$\begin{pmatrix} 0 & 2 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix}.$$

Nech G je bezslučkový [orientovaný] graf s n vrcholmi a m hranami. **Maticou incidencie** [orientovaného] grafu G pri usporiadaní jeho vrcholov v_1, \dots, v_n a hrán h_1, \dots, h_m je matica

$$(a_{ij})_{\substack{i \in \{1, \dots, n\} \\ j \in \{1, \dots, m\}}},$$

kde v prípade orientovaného grafu

$$a_{ij} = \begin{cases} 1 & , \text{ ak } v_i \text{ je počiatkový vrchol hrany } h_j \\ -1 & , \text{ ak } v_i \text{ je koncový vrchol hrany } h_j \\ 0 & , \text{ ak } v_i \text{ neinciduje s hranou } h_j \end{cases}$$

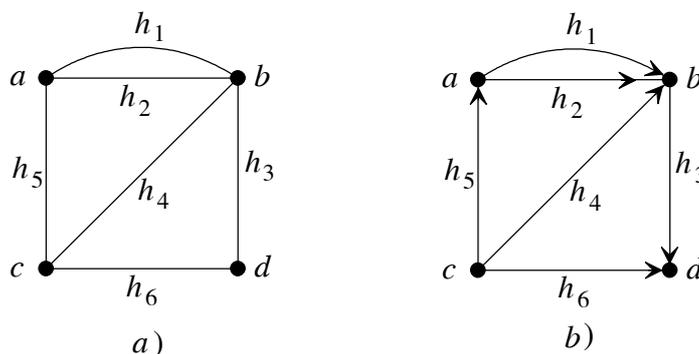
a v prípade neorientovaného grafu

$$a_{ij} = \begin{cases} 1 & , \text{ ak } v_i \text{ inciduje s } h_j \\ 0 & , \text{ ak } v_i \text{ neinciduje s } h_j \end{cases} .$$

Príklad.

a) Maticou incidencie grafu z obr. 41a pri usporiadaní vrcholov a, b, c, d a hrán $h_1, h_2, h_3, h_4, h_5, h_6$ je matica

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



Obr. 41

b) Maticou incidencie orientovaného grafu z obr. 41b pri usporiadaní vrcholov a, b, c, d a hrán $h_1, h_2, h_3, h_4, h_5, h_6$ je matica

$$\begin{pmatrix} 1 & 1 & 0 & 0 & -1 & 0 \\ -1 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & -1 & -1 & 0 & -1 \end{pmatrix} .$$

Veta. Matica susednosti neorientovaného grafu je symetrická matica.

Dôkaz. Pre prvky m_{ij} matice susednosti platí

$$m_{ij} = m(v_i, v_j) = m(v_j, v_i) = m_{ji} .$$

To však znamená, že táto matica je symetrická.



Veta. Nech G je bezslučkový [orientovaný] graf so zvoleným usporiadaním jeho vrcholov v_1, \dots, v_n a hrán h_1, \dots, h_m . Potom pre maticu incidencie $(a_{ij})_{\substack{i \in \{1, \dots, n\} \\ j \in \{1, \dots, m\}}}$ [orientovaného] grafu G platí

$$\sum_{j=1}^m |a_{ij}| = st(v_i).$$

Dôkaz. Podľa definície matice incidencie je $|a_{ij}| = 1$ práve vtedy, keď vrchol v_i inciduje s hranou h_j . Stupeň vrcholu bezslučkového [orientovaného] grafu sa rovná počtu hrán, s ktorými tento vrchol inciduje. Z týchto dvoch faktov už vyplýva tvrdenie vety. \square

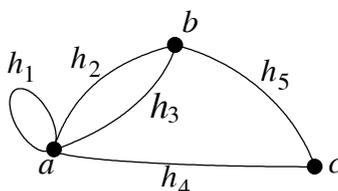
Poznámka. Maticu incidencie je možné definovať aj pre [orientované] grafy obsahujúce slučky. Stačí k definícii pridať:

Ak h_j je slučka, ktorá inciduje s vrcholom v_k , tak $a_{kj} = 1$ a pre všetky $i \neq k$ je $a_{ij} = 0$.

Predchádzajúca veta však pre [orientované] grafy so slučkou neplatí.

Príklad. Maticou incidencie grafu z obr. 42 pri usporiadaní vrcholov a, b, c a hrán h_1, h_2, h_3, h_4, h_5 je matica

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



Obr. 42

Cvičenia

1. Čo platí o [orientovanom] grafe, ktorého matica susednosti má na diagonále len číslo 0?
2. Čo platí o [orientovanom] grafe, ktorého matica incidencie má niektorý riadok nulový?
3. Nakreslite diagram grafu reprezentovaného maticou susednosti typu 7×7 . Jej prvky $m_{ij} = 1$ práve vtedy, keď $i + 1$ delí $j + 1$ alebo $j + 1$ delí $i + 1$. V opačnom prípade je $m_{ij} = 0$.

Sledy

Nech u, v sú vrcholy [orientovaného] grafu G , $n \in \mathbb{N}$. **Sledom** dĺžky n z vrcholu u do vrcholu v nazývame postupnosť

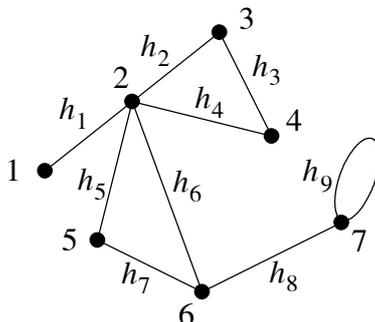
$$(v_0, h_1, v_1, h_2, v_2, \dots, v_{n-1}, h_n, v_n),$$

kde

1. $v_0 = u, v_n = v$;
 2. v_0, v_1, \dots, v_n sú vrcholy [orientovaného] grafu G ;
 3. h_1, \dots, h_n sú hrany [orientovaného] grafu G ;
 4. pre $i \in \{1, \dots, n\}$ hrana h_i inciduje s vrcholmi v_{i-1}, v_i .
- Vrcholy u, v sa nazývajú **krajné vrcholy** tohto **sledu**.

Príklad. V grafe na obr. 43

- a) $(1, h_1, 2, h_6, 6, h_7, 5, h_5, 2, h_2, 3)$ je sled dĺžky 5 z vrcholu 1 do vrcholu 3;
- b) (5) je sled dĺžky 0 z vrcholu 5 do vrcholu 5;
- c) $(7, h_9, 7)$ je sled dĺžky 1 z vrcholu 7 do vrcholu 7.



Obr. 43

Nech u, v sú vrcholy orientovaného grafu G , $n \in \mathbb{N}$. **Orientovaným sledom** dĺžky n z vrcholu u do vrcholu v nazývame postupnosť

$$(v_0, h_1, v_1, h_2, v_2, \dots, v_{n-1}, h_n, v_n),$$

kde

1. $v_0 = u, v_n = v$;
 2. v_0, v_1, \dots, v_n sú vrcholy orientovaného grafu G ;
 3. h_1, \dots, h_n sú hrany [orientovaného] grafu G ;
 4. pre $i \in \{1, \dots, n\}$ je v_{i-1} počiatkový a v_i koncový vrchol hrany h_i .
- Vrchol u sa nazýva **počiatkový** a vrchol v **koncový vrchol** tohoto **sledu**.

Poznámka. V súvislosti s [orientovaným] sledom $(v_0, h_1, \dots, h_{i-1}, v_i, h_i, \dots, h_n, v_n)$ v [orientovanom] grafe budeme tiež hovoriť, že hrana h_{i-1} **vchádza** do vrcholu v_i a hrana h_i z vrcholu v_i **vychádza**.

V jednoduchých [orientovaných] grafoch je hrana jednoznačne určená jej [počiatočným a koncovým vrcholom] krajnými vrcholmi. Preto v jednoduchých [orientovaných] grafoch budeme [orientovaný] sled

$$(v_0, h_1, v_1, h_2, v_2, \dots, v_{n-1}, h_n, v_n)$$

zapisovať len ako postupnosť vrcholov

$$(v_0, v_1, v_2, \dots, v_{n-1}, v_n).$$

Podobne, [orientovanú] hranu [s počiatočným vrcholom x a koncovým y] s krajnými vrcholmi x, y budeme označovať (x, y) .

[Orientovaný] sled z vrcholu u do vrcholu v sa nazýva **uzavretý**, ak $u = v$.
V opačnom prípade sa nazýva **otvorený**.

Veta. *Nech A je matica susednosti [orientovaného] grafu G pri usporiadaní jeho vrcholov v_1, v_2, \dots, v_n . Potom prvok v i -tom riadku a j -tom stĺpci, $i, j \in \{1, \dots, n\}$, matice A^k , $k \in \mathbb{N}$, je počet všetkých [orientovaných] sledov dĺžky k z vrcholu v_i do vrcholu v_j .*

Dôkaz. (Matematickou indukciou vzhľadom na k)

V prípade $k = 0$, je A^k jednotková matica. Z druhej strany, [orientované] sledy dĺžky 0 sú len [orientované] sledy (v_i) , to znamená, že počty [orientovaných] sledov dĺžky 0 z vrcholu v_i do vrcholu v_j sú čísla

$$b_{ij} = \begin{cases} 1, & \text{pre } i = j \\ 0, & \text{pre } i \neq j \end{cases},$$

čo sú práve prvky jednotkovej matice.

Predpokladajme, že tvrdenie vety platí pre $k - 1 \geq 0$. Pre $i, j \in \{1, \dots, n\}$ označme a_{ij} prvky matice A , b_{ij} prvky matice A^{k-1} a c_{ij} prvky matice A^k . Potom z vlastnosti

$$A^k = AA^{k-1}$$

vyplýva

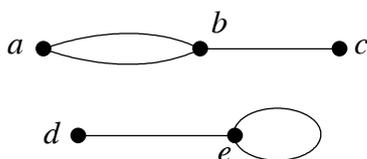
$$c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \dots + a_{in}b_{nj} = \sum_{r=1}^n a_{ir}b_{rj}.$$

Číslo a_{ir} sa podľa definície matice susednosti rovná počtu všetkých [orientovaných] hrán [s počiatočným vrcholom v_i a koncovým v_r] s krajnými vrcholmi v_i, v_r , t.j. [orientovaných] sledov dĺžky 1 z vrcholu v_i do vrcholu v_r . Podľa indukčného predpokladu je zas b_{rj} počet všetkých [orientovaných] sledov dĺžky $k - 1$ z vrcholu v_r do vrcholu v_j . Potom však číslo $a_{ir}b_{rj}$ určuje počet všetkých [orientovaných] sledov dĺžky k z vrcholu v_i do vrcholu v_j , v ktorých druhým vrcholom je v_r . Súčtom všetkých týchto čísel dostaneme počet všetkých [orientovaných] sledov dĺžky k z vrcholu v_i do vrcholu v_j . 

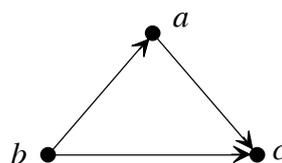
[Orientovaný] graf G sa nazýva **súvislý**, ak pre každé dva vrcholy u, v [orientovaného] grafu G existuje sled z vrcholu u do vrcholu v .

Orientovaný graf G sa nazýva **silne súvislý**, ak pre každé dva vrcholy u, v orientovaného grafu G existuje orientovaný sled z vrcholu u do vrcholu v .

Príklad. Graf na obr. 44 nie je súvislý, lebo napr. neexistuje sled z vrcholu a do vrcholu d . Orientovaný graf na obr. 45 je súvislý, ale nie je silne súvislý, lebo napr. neexistuje orientovaný sled z a do b . ■



Obr. 44



Obr. 45

Na množine všetkých vrcholov [orientovaného] grafu $G = (V, E, e)$ definujeme reláciu ω takto:

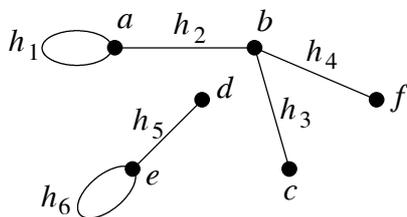
$$u \omega v \Leftrightarrow \text{existuje sled s krajnými vrcholmi } u, v.$$

Lahko sa presvedčíte, že ω je reláciou ekvivalencie na množine V . Pomocou tried ekvivalencie relácie ω (ony tvoria rozklad množiny V) vytvoríme podgrafy [orientovaného] grafu G :

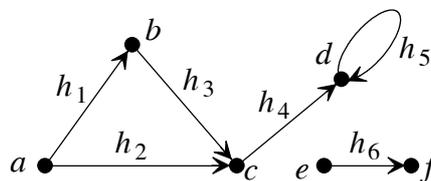
Nech U je trieda ekvivalencie relácie ω . Indukovaný podgraf $G(U)$ [orientovaného] grafu G sa nazýva **komponent súvislosti [orientovaného] grafu G** .

Poznámka. Ak U je trieda ekvivalencie vrcholu u , čiže $U = \omega(u)$, tak hovoríme tiež, že $G(U)$ je komponent súvislosti grafu G obsahujúci vrchol u .

Príklad. Komponent súvislosti grafu G na obr. 46 obsahujúci vrchol c je podgraf $G(\omega(c))$, ktorého množina vrcholov je $\omega(c) = \{a, b, c, f\}$ a množina hrán je $\{h_1, h_2, h_3, h_4\}$. Graf G má dva komponenty súvislosti.



Obr. 46



Obr. 47

Komponent súvislosti orientovaného grafu H na obr. 47 obsahujúci vrchol c je jeho podgraf $H(\omega(c))$, ktorého množina vrcholov je $\omega(c) = \{a, b, c, d\}$ a množina hrán je $\{h_1, h_2, h_3, h_4\}$. ■

Veta. Komponent súvislosti [orientovaného] grafu G je súvislý [orientovaný] graf.

Dôkaz, pre jeho jednoduchosť, prenechávame čitateľovi. 🔔

Veta. *Nech G' je súvislý podgraf [orientovaného] grafu G . Potom G' je podgrafom práve jedného komponentu súvislosti grafu G .*

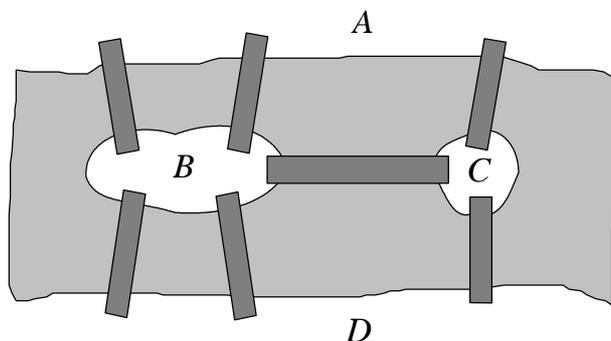
Dôkaz. Nech u je vrchol G' . Ukážeme, že G' je podgrafom komponentu súvislosti $G(\omega(u))$. Na to stačí dokázať, že ľubovoľný vrchol a ľubovoľná hrana z G' patrí do $G(\omega(u))$. Nech teda v je vrchol [orientovaného] grafu G' . Zo súvislosti tohto [orientovaného] grafu vyplýva, že v ňom existuje sled s krajnými vrcholmi u, v . To však znamená, že vrchol v patrí do triedy ekvivalencie $\omega(u)$, a teda je vrcholom komponentu súvislosti $G(\omega(u))$. Nech h je hrana [orientovaného] grafu G' incidentná s vrcholmi x, y . Z predošlej časti dôkazu vyplýva, že vrcholy x, y sú tiež vrcholmi komponentu súvislosti $G(\omega(u))$. Hrana h je zároveň aj hranou [orientovaného] grafu G . Keďže indukovaný podgraf $G(\omega(u))$ [orientovaného] grafu G obsahuje všetky hrany [orientovaného] grafu G , ktorých krajné vrcholy sú z podgrafu $G(\omega(u))$, je h hranou komponentu súvislosti $G(\omega(u))$.

Ešte treba ukázať, že G' nie je podgrafom iného komponentu súvislosti. To však vyplýva z toho, že dva rôzne komponenty súvislosti [orientovaného] grafu G nemajú spoločný žiadny vrchol, lebo množiny ich vrcholov sú rôznymi triedami rozkladu množiny vrcholov [orientovaného] grafu G . 🔔

[Orientovaný] sled, v ktorom sa každá hrana vyskytuje najviac raz, sa nazýva [orientovaný] *tah*.

[Orientovaný] sled, v ktorom sa každý vrchol vyskytuje najviac raz, sa nazýva [orientovaná] *cesta*.

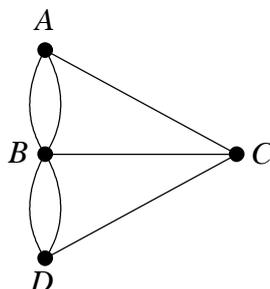
Príklad. V prvej práci z teórie grafov, ktorá pochádza od L. Eulera z roku 1736, sa nachádza riešenie zaujímavého problému, známeho ako problém königsbergských mostov. Mesto Königsberg (dnešný Kaliningrad v Rusku) leží na oboch brehoch rieky Pregel (Pregolja) a dvoch ostrovoch tejto rieky. Tieto ostrovy sú vzájomne a tiež s brehmi rieky pospájané mostami, ako to vidieť na obr. 48. Spomenutý problém je vyjsť z ľubovoľného miesta A, B, C alebo D, prejsť každým mostom práve raz a skončiť vo východiskovom mieste.



Obr. 48

Celý problém je možné modelovať na grafe z obr. 49, kde vrcholy grafu predstavujú brehy A, B a ostrovy C, D a hrany predstavujú mosty. V jazyku grafov môžeme tento problém

formulovať takto: nájsť v tomto grafe uzavretý ťah, ktorý obsahuje všetky vrcholy a všetky hrany.



Obr. 49

Neskôr ukážeme, že takýto ťah v tomto grafe neexistuje. ■

Ťah grafu G sa nazýva **eulerovský**, ak obsahuje všetky vrcholy a všetky hrany grafu G .

Graf G sa nazýva **eulerovský**, ak v ňom existuje uzavretý eulerovský ťah.

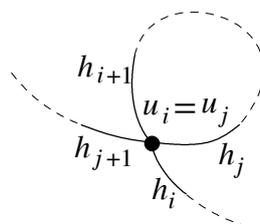
Veta. Graf G je eulerovský práve vtedy, keď je súvislý a všetky jeho vrcholy sú párneho stupňa.

Dôkaz. Nech G je eulerovský graf. Predpokladajme najprv, že graf G nemá žiadnu hranu a že u je vrchol grafu G . Potom jediným uzavretým ťahom obsahujúcim vrchol u je ťah (u) . Tento ťah musí byť eulerovský, a teda obsahuje všetky vrcholy grafu G . Preto u je jediným vrcholom grafu G . G je teda súvislý graf a $st(u) = 0$, čo je párne číslo.

Nech teraz graf G má aspoň jednu hranu a nech

$$T = (u = u_0, h_1, u_1, \dots, u_{n-1}, h_n, u_n = u)$$

je uzavretý eulerovský ťah z vrcholu u do vrcholu u . Nech vrchol u_i je rôzny od vrcholu u (i je určite rôzne od 0 a od n). Potom v tomto ťahu hrana h_i vchádza do vrcholu u_i a hrana h_{i+1} z neho vychádza. Ak vrchol u_i sa v eulerovskom ťahu T viackrát nevyskytuje, jeho stupeň je 2. Ak ťah T opätovne prechádza vrcholom u_i , tak potom existuje $j \neq i$ tak, že $u_j = u_i$, hrana h_j do vrcholu u_i vchádza a hrana h_{j+1} z neho vychádza (obr. 50).



Obr. 50

Pre čísla i, j môžu nastať tri prípady:

1. $j = i + 1$,
2. $i = j + 1$,

3. $|i - j| > 1$.

V prvom prípade h_i je slučka a v druhom prípade h_j je slučka. V treťom prípade sú hrany $h_{i-1}, h_i, h_{j-1}, h_j$ navzájom rôzne. Vo všetkých troch prípadoch hrany $h_{i-1}, h_i, h_{j-1}, h_j$ zvyšujú stupeň vrcholu u_i o 4. To znamená, že každý ďalší výskyt vrcholu u_i v eulerovskom ťahu T zvyšuje stupeň tohto vrcholu o 2. Teda ak v eulerovskom ťahu sa vrchol u_i vyskytuje k -krát, tak stupeň vrcholu u_i je $2k$. V prípade vrcholu u je situácia analogická. Ak sa vrchol u vyskytuje len na začiatku a na konci eulerovského ťahu T , tak stupeň vrcholu u je zrejme 2. Každý ďalší výskyt vrcholu u v T zvyšuje stupeň vrcholu u o 2. Tým sme dokázali, že stupeň každého vrcholu eulerovského grafu je párny.

Dokážeme ešte, že G je súvislý graf. Nech v, w sú dva rôzne vrcholy grafu G . Obidva sa musia nachádzať v eulerovskom ťahu T . Nech teda $v = u_i, w = u_j$. Potom v prípade $i < j$,

$$(v = u_i, h_i, u_{i+1}, h_{i+1}, \dots, h_{j-1}, u_j = w)$$

je sled z vrcholu v do vrcholu w . Ak $j < i$, je

$$(w = u_j, h_j, u_{j+1}, h_{j+1}, \dots, h_{i-1}, u_i = v)$$

sled z vrcholu w do vrcholu v . To však znamená, že graf G je súvislý.

Opačnú implikáciu, t.j. ak graf G je súvislý a stupeň každého jeho vrcholu je párny, tak G je eulerovský graf, dokážeme matematickou indukciou vzhľadom na počet hrán. Nech teda G je súvislý graf, ktorý má n hrán a stupeň každého jeho vrcholu je párny.

V prípade $n = 0$, má súvislý graf G jediný vrchol, ktorého stupeň je 0. Ak tento vrchol označíme u , tak (u) je uzavretý eulerovský ťah. To však znamená, že G je eulerovský graf.

V prípade $n = 1$, má graf G jedinú hranu h , ktorá musí incidovať len s jedným vrcholom (v opačnom prípade by graf obsahoval dva vrcholy stupňa 1), označme ho u . Zo súvislosti grafu G vyplýva, že viac vrcholov nemôže mať. Potom však (u, h, u) je uzavretý eulerovský ťah, a teda G je eulerovský graf.

Predpokladajme, že $n \geq 2$ a že tvrdenie platí pre grafy s k hranami, $k < n$. V súvislom grafe s aspoň dvoma hranami existujú dve rôzne hrany, ktoré incidujú s tým istým vrcholom. Nech teda h_1, h_2 sú rôzne hrany grafu G , ktoré incidujú s jedným vrcholom, označme ho v . Utvorme nový graf G' , ktorý vznikne z grafu G odobraním niektorej hrany a prípadne odobraním niektorého vrcholu a pridaním novej hrany. Ak niektorá z hrán h_1, h_2 je slučka, graf G' vznikne z G odobraním tejto slučky (ak obe hrany sú slučky, tak odoberieme len jednu z nich). Ak žiadna z hrán h_1, h_2 nie je slučka, tak okrem vrcholu v , hrana h_1 inciduje s vrcholom $v_1, v_1 \neq v$ a hrana h_2 inciduje s vrcholom $v_2, v_2 \neq v$. V tomto prípade graf G' vznikne z G odobraním hrán h_1, h_2 , pridaním novej hrany h , ktorá inciduje s vrcholmi v_1, v_2 a v prípade, že vrchol v neinciduje s ďalšou linkou rôznou od liniek h_1, h_2 odobraním vrcholu v a všetkých s ním incidujúcich slučiek. V oboch prípadoch je G' súvislý graf s počtom hrán menším ako $n - 1$, v ktorom každý vrchol je párneho stupňa. Podľa indukčného predpokladu je G' eulerovský graf, teda v ňom existuje uzavretý eulerovský ťah. Pozmeňme tento ťah v závislosti od toho, ako vznikol graf G' . Ak to bolo odobraním slučky $h_i, i \in \{1, 2\}$, tak pridajme do eulerovského ťahu grafu G' za prvý výskyt vrcholu v dva členy h_i, v . Ak to bolo odobraním oboch hrán h_1, h_2 (vrchol v sme ponechali), nahraďme v uzavretom eulerovskom ťahu grafu G' hranu h tromi novými členmi h_1, v, h_2 . Ak to bolo odobraním hrán h_1, h_2 , vrcholu v a s ním incidujúcich slučiek g_1, \dots, g_r , nahraďme v uzavretom eulerovskom ťahu grafu G' hranu h novými členmi $h_1, v, g_1, v, g_2, \dots, g_n, v, h_2$. Vo všetkých prípadoch dostaneme

uzavretý ťah grafu G , ktorý obsahuje všetky hrany aj vrcholy tohto grafu, teda uzavretý eulerovský ťah. To znamená, že graf G je eulerovský. \triangle

Príklad. Vidíme, že graf na obr. 49 nemá všetky vrcholy párneho stupňa, preto v ňom neexistuje uzavretý eulerovský ťah. To znamená, že v Kaliningrade nemožno vystúpiť z jedného miesta, prejsť všetkými mostami práve raz a skončiť na tom istom mieste. \blacksquare

Veta. V grafe G existuje otvorený eulerovský ťah práve vtedy, keď G je súvislý graf, ktorý obsahuje práve dva vrcholy nepárneho stupňa.

Dôkaz. Nech $u \neq v$ a nech

$$(u = u_0, h_1, \dots, h_n, u_n = v)$$

je otvorený eulerovský ťah grafu G . Z grafu G vytvoríme nový graf G' pridaním novej hrany h , ktorá inciduje s vrcholmi u, v . Potom

$$(u = u_0, h_1, \dots, h_n, v, h, u)$$

je uzavretý eulerovský ťah grafu G' . Podľa predchádzajúcej vety stupne všetkých vrcholov grafu G' sú párne. Pretože grafy G' a G sa odlišujú len hranou h incidentnou s vrcholmi u, v , majú vrcholy u, v v grafe G stupne o 1 menšie ako v grafe G' , teda nepárne. Všetky ostatné vrcholy v grafe G majú rovnaké stupne ako v grafe G' , teda párne.

Predpokladajme teraz, že G je súvislý graf, ktorý obsahuje práve 2 vrcholy nepárneho stupňa. Nech sú to vrcholy u, v . Pridaním novej hrany h incidentnej s vrcholmi u, v dostaneme nový graf G' , ktorý je súvislý a všetky jeho vrcholy sú párneho stupňa. Graf G' je teda eulerovský a existuje v ňom uzavretý eulerovský ťah z vrcholu v do vrcholu v :

$$(v, h, u = u_0, h_1, \dots, h_n, u_n = v).$$

Potom

$$(u = u_0, h_1, \dots, h_n, u_n = v)$$

je otvorený eulerovský ťah grafu G . \triangle

Vzdialenosťou vrcholov u, v [orientovaného] grafu G nazývame minimum z dĺžok všetkých [orientovaných] sledov z vrcholu u do vrcholu v a označujeme $\rho(u, v)$.

Ak v [orientovanom] grafe G neexistuje [orientovaný] sled z u do v , kladieme $\rho(u, v) = \infty$.

Poznámka. Pre symbol ∞ zadefinujeme tieto vťahy: pre všetky $n \in \mathbb{N}$ je

$$n + \infty = \infty + n = \infty + \infty = \infty, \quad \infty > n.$$

Veta. Pre ľubovoľné vrcholy u, v, w grafu G platí

$$(a) \rho(u, v) \geq 0,$$

$$(b) \rho(u, v) = 0 \text{ práve vtedy, keď } u = v,$$

$$(c) \rho(u, v) = \rho(v, u),$$

$$(d) \rho(u, v) + \rho(v, w) \geq \rho(u, w).$$

Dôkaz.

(a) Podľa definície je $\rho(u, v)$ buď ∞ , alebo dĺžka niektorého sledu z vrcholu u do vrcholu v , čo je prirodzené číslo. V oboch prípadoch je $\rho(u, v) \geq 0$.

(b) Nech $\rho(u, v) = 0$. Potom existuje sled z u do v dĺžky 0. To je však možné len vtedy, keď $u = v$.

Naopak, ak $u = v$, tak (u) je sled dĺžky 0 z u do v . Sled z u do v menšej dĺžky nemôže existovať, preto $\rho(u, v) = 0$.

(c) Predpokladajme, že $\rho(u, v) \neq \rho(v, u)$. Nech teda $\rho(u, v) = r$, $\rho(v, u) = s$, $r < s$ a nech

$$(u = u_0, h_1, u_1, \dots, u_{r-1}, h_r, u_r = v)$$

je sled dĺžky r z vrcholu u do vrcholu v . Potom však

$$(v = u_r, h_r, u_{r-1}, \dots, u_1, h_1, u_0 = u)$$

je sled dĺžky r z vrcholu v do vrcholu u a podľa definície vzdialenosti vrcholov je

$$s = \rho(v, u) \leq r,$$

čo je v spore s predpokladom.

(d) Ak $\rho(u, v) = \infty$ alebo $\rho(v, w) = \infty$, tak $\rho(u, v) + \rho(v, w) = \infty$. Vzhľadom na to, že $\rho(u, w) \leq \infty$, je nerovnosť $\rho(u, v) + \rho(v, w) \geq \rho(u, w)$ v tomto prípade splnená.

Ak $\rho(u, v) = r < \infty$, $\rho(v, w) = s < \infty$, tak existuje sled $(u = u_0, h_1, \dots, h_r, u_r = v)$ dĺžky r z u do v a sled $(v = v_0, g_1, \dots, g_s, v_s = w)$ dĺžky s z v do w . Potom

$$(u, h_1, \dots, h_r, v, g_1, \dots, g_s, w)$$

je sled dĺžky $r + s$ z vrcholu u do vrcholu w . Preto

$$\rho(u, w) \leq r + s = \rho(u, v) + \rho(v, w).$$



Cvičenia

- Ukážte, že nesúvislý jednoduchý bezslučkový graf s n vrcholmi má maximálne $\frac{(n-1)(n-2)}{2}$ hrán.
- Ukážte, že počet všetkých sledov dĺžky n z vrcholu v_1 do vrcholu v_1 v grafe na obr. 51 sa rovná Fibonacciho číslu f_n , ktoré je definované takto:
 $f_0 = 1$, $f_1 = 1$, $f_n = f_{n-1} + f_{n-2}$ pre $n \in \mathbb{N}$, $n \geq 2$.



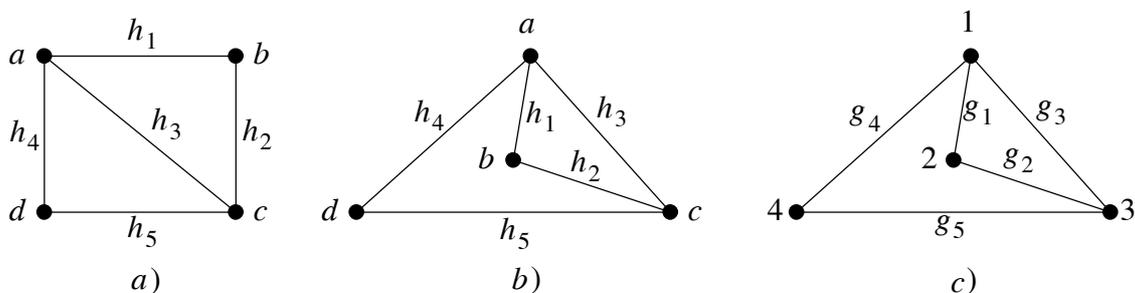
Obr. 51

- Ukážte, že v súvislom grafe s aspoň dvoma hranami, existujú dve hrany, ktoré incidujú s tým istým vrcholom.

4. Uvedte príklad orientovaného grafu, v ktorom neplatí pre všetky vrcholy u, v rovnosť $\rho(u, v) = \rho(v, u)$.
5. Nech S je sled z vrcholu u do vrcholu v , ktorého dĺžka je $\rho(u, v)$. Dokážte, že S je cesta.

Izomorfizmus [orientovaných] grafov

Všimnime si diagramy grafov G_1, G_2 a G_3 na obr. 52. Grafy G_1 a G_2 sú rovnaké (majú rovnaké množiny vrcholov, množiny hrán a tiež incidencie). Grafy G_1 a G_3 sú rôzne. Ale keď porovnáme ich diagramy na obr. 52b, 52c, vidíme, že medzi týmito grafmi nie je podstatný rozdiel, líšia sa iba v pomenovaní vrcholov a hrán. Takéto grafy budeme nazývať izomorfné.



Obr. 52

Presná definícia je takáto:

[**Orientované**] grafy $G = (V, E, e)$, $G' = (V', E', e')$ sa nazývajú **izomorfné**, ak existujú bijektívne zobrazenia $\varphi: V \rightarrow V'$, $\psi: E \rightarrow E'$ zachovávajúce incidenciu, t.j. pre každú hranu $h \in E$ platí (obr. 53)

$$[\text{ak } e(h) = (u, v), \text{ tak } e'(\psi(h)) = (\varphi(u), \varphi(v))]$$

$$\text{ak } e(h) = \{u, v\}, \text{ tak } e'(\psi(h)) = \{\varphi(u), \varphi(v)\}.$$

Usporiadaná dvojica (φ, ψ) sa nazýva **izomorfizmus** [orientovaného] grafu G na G' .

Príklad. Ukážeme, že pre grafy $G_1 = (V_1, E_1, e_1)$ a $G_3 = (V_3, E_3, e_3)$ z obr. 52a, 52c je dvojica zobrazení (φ, ψ) ,

$$\varphi: V_1 \rightarrow V_3, \quad \varphi(a) = 1, \varphi(b) = 2, \varphi(c) = 3, \varphi(d) = 4,$$

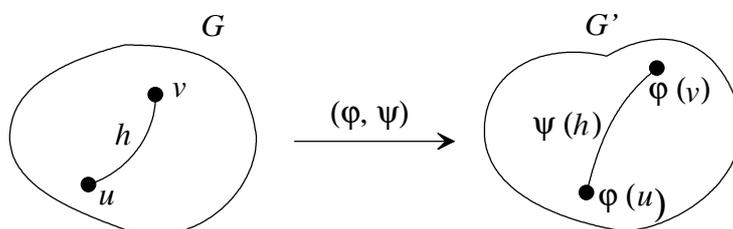
$$\psi: E_1 \rightarrow E_3, \quad \psi(h_i) = g_i, \quad i \in \{1, 2, 3, 4\}$$

izomorfizmus.

Riešenie. Zobrazenia φ a ψ sú zrejme bijekcie. Stačí ukázať, že zachovávajú incidenciu: $e_1(h_1) = \{a, b\}$, $e_3(\psi(h_1)) = e_3(g_1) = \{1, 2\} = \{\varphi(a), \varphi(b)\}$,
 $e_1(h_2) = \{b, c\}$, $e_3(\psi(h_2)) = e_3(g_2) = \{2, 3\} = \{\varphi(b), \varphi(c)\}$,

$$\begin{aligned}
 e_1(h_3) &= \{a, c\}, & e_3(\psi(h_3)) &= e_3(g_3) = \{1, 3\} = \{\varphi(a), \varphi(c)\}, \\
 e_1(h_4) &= \{a, d\}, & e_3(\psi(h_4)) &= e_3(g_4) = \{1, 4\} = \{\varphi(a), \varphi(d)\}, \\
 e_1(h_5) &= \{c, d\}, & e_3(\psi(h_5)) &= e_3(g_5) = \{4, 3\} = \{\varphi(d), \varphi(c)\}.
 \end{aligned}$$

■



Obr. 53

Veta. Nech $G = (V, E, e)$, $G' = (V', E', e')$ sú jednoduché [orientované] grafy. Potom nasledujúce tvrdenia sú ekvivalentné:

- G a G' sú izomorfné.
- Existuje bijekcia $\varphi : V \rightarrow V'$ tak, že pre všetky $u, v \in V$ platí: [u je počiatočný a v koncový vrchol jednej hrany práve vtedy, keď $\varphi(u)$ je počiatočný a $\varphi(v)$ koncový vrchol jednej hrany.] vrcholy u, v sú susedné práve vtedy, keď sú susedné vrcholy $\varphi(u), \varphi(v)$.

Dôkaz. Urobíme ho len pre neorientované grafy. Dôkaz pre orientované grafy je analogický.

Priamo z definície izomorfizmu je zrejmé, že z (a) vyplýva (b). Dokážeme implikáciu (b) \Rightarrow (a). Nech teda $\varphi : V \rightarrow V'$ je bijekcia splňujúca podmienku: vrcholy u, v grafu G sú susedné práve vtedy, keď sú susedné vrcholy $\varphi(u), \varphi(v)$ grafu G' . Keďže sa zaoberáme len jednoduchými grafmi, budeme hrany zapisovať ako usporiadané dvojice krajných vrcholov. Uvedená vlastnosť zobrazenia φ nám umožňuje definovať zobrazenie medzi množinami hrán grafov G a G' týmto spôsobom:

$$\psi : E \rightarrow E', \quad \psi(u, v) = (\varphi(u), \varphi(v)).$$

Keďže φ je bijekcia a (u, v) je hrana v G práve vtedy, keď $(\varphi(u), \varphi(v))$ je hrana v G' , je ψ tiež bijekcia. Pre bijekcie φ, ψ platí

$$e(u, v) = \{u, v\}, \quad e'(\psi(u, v)) = e'(\varphi(u), \varphi(v)) = \{\varphi(u), \varphi(v)\}$$

z čoho vyplýva, že (φ, ψ) je izomorfizmus, a teda grafy G, G' sú izomorfné. 🔔

Predchádzajúcu vetu je možné zovšeobecniť aj pre multigrafy.

Veta. [Orientované] grafy $G = (V, E, e)$, $G' = (V', E', e')$ sú izomorfné práve vtedy, keď existuje bijekcia $\varphi : V \rightarrow V'$ a pre každé vrcholy u, v [orientovaného] grafu G je

$$[m^+(u, v) = m^+(\varphi(u), \varphi(v))]$$

$$m(u, v) = m(\varphi(u), \varphi(v)).$$

Dôkaz. Vetu dokážeme pre orientované grafy. Pre neorientované grafy je dôkaz podobný.

Nech teda orientované grafy G a G' sú izomorfné. To znamená, že existujú také bijekcie $\varphi : V \rightarrow V'$, $\psi : E \rightarrow E'$, že pre každú hranu $h \in E$ platí

$$\text{ak } e(h) = (x, y), \text{ tak } e'(\psi(h)) = (\varphi(x), \varphi(y)).$$

Nech $u, v \in V$ a $m^+(u, v) = k$. Ak $k = 0$, znamená to, že v grafe G neexistuje hrana s počiatočným vrcholom u a koncovým v . Potom ani v grafe G' neexistuje hrana s počiatočným vrcholom $\varphi(u)$ a koncovým $\varphi(v)$, v opačnom prípade by ψ nebola bijekcia. Ak $k \geq 1$, tak existuje k orientovaných hrán h_1, \dots, h_k , pre ktoré je

$$e(h_i) = (u, v), \quad i \in \{1, \dots, k\}.$$

Vzhľadom na to, že

$$e'(\psi(h_i)) = (\varphi(u), \varphi(v)), \quad \text{pre } i \in \{1, \dots, k\}$$

a ψ je bijekcia, je $m^+(\varphi(u), \varphi(v)) = k$.

Dokážeme opačnú implikáciu. Nech teda existuje bijekcia $\varphi : V \rightarrow V'$ tak, že pre všetky $x, y \in V$ je $m^+(x, y) = m^+(\varphi(x), \varphi(y))$. Nech u je počiatočný a v koncový vrchol niektorej orientovanej hrany grafu G a nech $m^+(u, v) = k \geq 1$. Existuje teda práve k orientovaných hrán h_1, \dots, h_k s počiatočným vrcholom u a koncovým v . Keďže $m^+(\varphi(u), \varphi(v)) = m^+(u, v)$, existuje v grafe G' k orientovaných hrán g_1, \dots, g_k s počiatočným vrcholom $\varphi(u)$ a koncovým $\varphi(v)$. To nám umožňuje definovať zobrazenie ψ takto:

$$\psi : E \rightarrow E', \quad \psi(h_i) = g_i \quad \text{pre } i \in \{1, \dots, k\}.$$

Lahko sa ukáže (prenechávame to čitateľovi), že ψ je bijektívne zobrazenie a že zobrazenia φ a ψ zachovávajú incidenciu. Grafy G a G' sú teda izomorfné. 

Z tejto vety bezprostredne vyplývajú ďalšie dve.

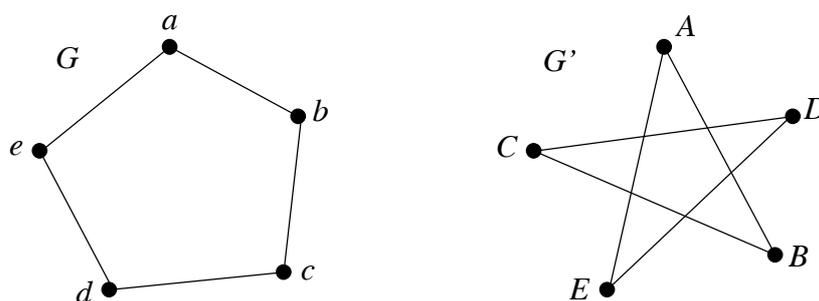
Veta. [Orientované] grafy G a G' sú izomorfné práve vtedy, keď existujú také usporiadania ich vrcholov, pri ktorých sú matice susednosti týchto [orientovaných] grafov rovnaké. 

Veta. [Orientované] grafy G a G' sú izomorfné práve vtedy, keď existujú také usporiadania ich vrcholov a hrán, pri ktorých sú matice incidencie týchto [orientovaných] grafov rovnaké. 

Príklad. Grafy G a G' na obr. 54 sú izomorfné, lebo matica susednosti grafu G pri usporiadaní vrcholov a, b, c, d, e je

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{pmatrix},$$

čo je zároveň aj matica susednosti grafu G' pri usporiadaní vrcholov A, B, C, D, E . ■



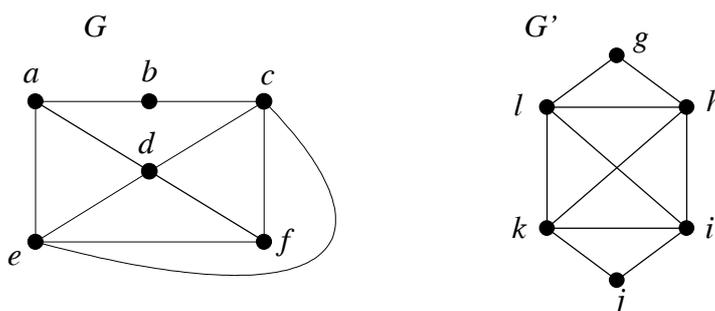
Obr. 54

Vlastnosť P sa nazýva **invariant [orientovaných] grafov**, ak pre každé izomorfné [orientované] grafy G a G' platí: ak G má vlastnosť P , tak aj G' má vlastnosť P .

Príklad. Ukážte, že grafy G a G' na obr. 55 nie sú izomorfné.

Riešenie. Najprv ukážeme, že vlastnosť „má vrchol stupňa k “, kde $k \in \mathbb{N}$, je invariant.

Nech grafy G_1 a G_2 sú izomorfné a (φ, ψ) je izomorfizmus z G_1 na G_2 . Nech graf G_1 má vrchol v stupňa k . Potom existuje k hrán h_1, \dots, h_k , ktoré incidujú s vrcholom v (slučka sa v tomto zozname uvádza dvakrát). Podľa definície izomorfizmu hrany $\psi(h_1), \dots, \psi(h_k)$ incidujú s vrcholom $\varphi(v)$ a keďže ψ je bijekcia, je $st(\varphi(v)) \geq k$. Nech H je hrana grafu G_2 , ktorá inciduje s vrcholom $\varphi(v)$. Keďže ψ je surjektívne zobrazenie, existuje hrana h grafu G_1 tak, že $\psi(h) = H$. Pretože hrana $\psi(h)$ inciduje s vrcholom $\varphi(v)$, musí podľa definície izomorfizmu hrana h incidovať s vrcholom v . To však znamená, že h je niektorá z hrán h_1, \dots, h_k , a teda H je niektorá z hrán $\psi(h_1), \dots, \psi(h_k)$. Preto $st(\varphi(v)) = k$.



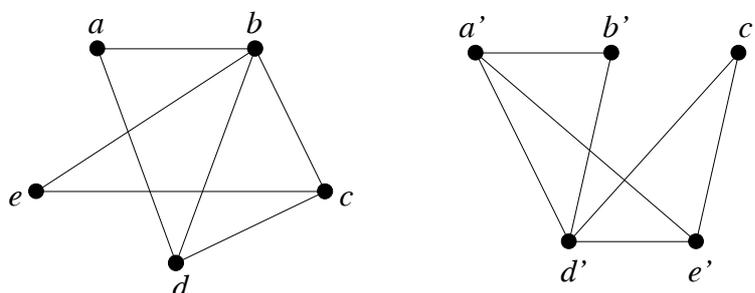
Obr. 55

Teraz už môžeme tvrdiť, že grafy G a G' nie sú izomorfné, lebo graf G má vrcholy stupňa 3 (a a f) a graf G' nemá vrcholy stupňa 3. ■

Cvičenia

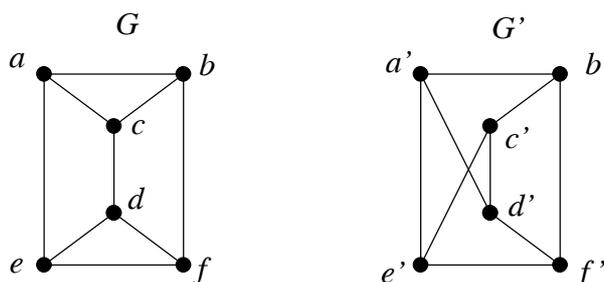
- Ukážte, že grafy G a G' na obr. 56 sú izomorfné a nájdite izomorfizmus grafu G na G' .
- Ukážte, že nasledujúce vlastnosti sú invarianty:
 - má n vrcholov stupňa k ,
 - má uzavretý ťah dĺžky k ,

- c) je súvislý,
 d) má hranu, ktorej jeden krajný vrchol má stupeň i a druhý má stupeň j .



Obr. 56

3. Ukážte, že nasledujúce vlastnosti sú invarianty orientovaných grafov:
 a) má vrchol, ktorého vstupný stupeň je k ,
 b) má vrchol, ktorého výstupný stupeň je k ,
 c) má uzavretý orientovaný sled,
 d) je silne súvislý.
4. Ukážte, že grafy G a G' na obr. 57 nie sú izomorfné.



Obr. 57

Stromy

Nech u je vrchol grafu G . Ťah nenulovej dĺžky z vrcholu u do vrcholu u sa nazýva **cyklus** z vrcholu u do vrcholu u .

Cyklus z vrcholu u do vrcholu u , v ktorom sa vrchol u vyskytuje práve dvakrát a ostatné vrcholy z G najviac raz, sa nazýva **jednoduchý cyklus**.

Veta. Ak v grafe G existuje cyklus z vrcholu u do vrcholu u , tak v G existuje jednoduchý cyklus z vrcholu u do vrcholu u .

Dôkaz. Nech cyklus

$$(u = u_0, h_1, u_1, \dots, h_n, u_n = u)$$

z vrcholu u do vrcholu u nie je jednoduchý. Potom existuje vrchol u_i , pre $i \in \{1, \dots, n-1\}$, ktorý sa v tomto cykle vyskytuje aspoň dvakrát. To ale znamená, že existuje také $j \in \{1, \dots, n\}$, že $j \neq i$, $u_i = u_j$. Keď z tohto cyklu odstránime úsek u_i, h_{i+1}, \dots, h_j (predpokladáme, že $i < j$), dostaneme opäť cyklus

$$(u = u_0, h_1, u_1, \dots, h_i, u_i, h_{j+1}, u_{j+1}, \dots, h_n, u_n = u),$$

v ktorom sa výskyt vrcholu u_i znížil o 1. Je zrejmé, že opakovaním tohto postupu konečný počet krát, získame cyklus, v ktorom sa každý vrchol z pôvodného cyklu, okrem vrcholu u , vyskytuje najviac raz. Tento cyklus je jednoduchý cyklus z u do u . \square

Veta. Nech u, v sú dva rôzne vrcholy grafu G a nech existujú dve rôzne cesty z vrcholu u do vrcholu v . Potom v grafe G existuje cyklus.

Dôkaz urobíme pre jednoduchý graf. Nech

$$(u = u_0, u_1, \dots, u_p = v) \text{ a } (u = v_0, v_1, \dots, v_q = v)$$

sú dve rôzne cesty z vrcholu u do vrcholu v . Označme r najmenšie číslo $i \in \{1, \dots, p\}$, pre ktoré platí $u_i \neq v_i$. Potom $u_{r-1} = v_{r-1}$ a $u_r \neq v_r$. Nech s je najmenšie z čísel $i \in \{r+1, \dots, p\}$, pre ktoré vrchol u_i sa rovná niektorému z vrcholov v_j , $j \in \{r+1, \dots, q\}$. Povedzme, že $u_s = v_t$. Potom

$$(u_{r-1}, u_r, \dots, u_s = v_t, v_{t-1}, \dots, v_r, v_{r-1} = u_{r-1})$$

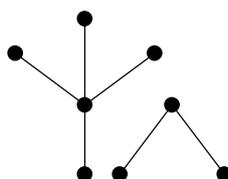
je jednoduchý cyklus. \square

Graf, v ktorom neexistuje cyklus, sa nazýva **acyklický**.

Príklad. Graf na obr. 58 je acyklický. \blacksquare

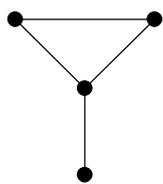
Jednoduchý bezslučkový graf G , v ktorom pre každé jeho dva vrcholy u, v existuje jediná cesta z u do v , sa nazýva **strom**.

Koreňový strom je strom s pevne vybraným vrcholom, ktorý nazývame **koreň**.

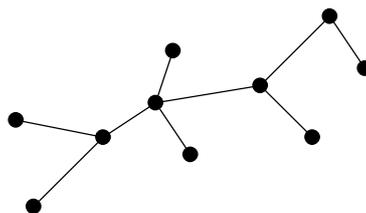


Obr. 58

Príklad. Lahko sa presvedčíte, že grafy z obr. 58 a 59 nie sú stromy a graf z obr. 60 je strom. \blacksquare



Obr. 59

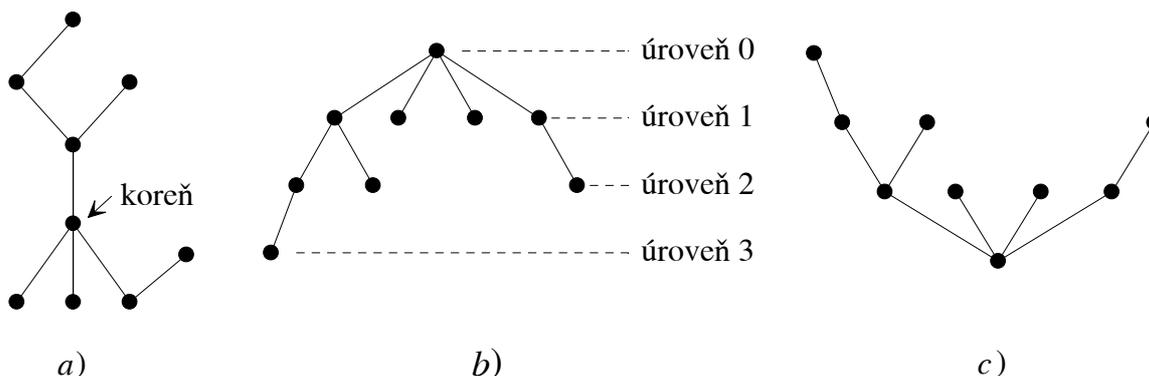


Obr. 60

Úrovníou vrcholu u v koreňovom strome s koreňom v , nazývame dĺžku cesty z v do u , t.j. $\rho(u, v)$. Maximum z úrovni všetkých vrcholov koreňového stromu sa nazýva **výška koreňového stromu**.

Diagram koreňového stromu sa spravidla kreslí tak, že vrcholy vyššej úrovne sa kreslia pod (resp. nad) vrcholmi nižšej úrovne.

Príklad. Koreňový strom z obr. 61a kreslíme tak, ako je to na obr. 61b resp. 61c. ■



Obr. 61

Veta. Pre graf S s $n \in \mathbb{N}^+$ vrcholmi sú nasledujúce tvrdenia ekvivalentné.

- (a) S je strom.
- (b) S je súvislý a acyklický graf.
- (c) S je súvislý graf a má $n - 1$ hrán.
- (d) S je acyklický graf a má $n - 1$ hrán.

Dôkaz. Aby sme dokázali ekvivalenciu tvrdení (a) až (d), stačí dokázať štyri implikácie: (a) \Rightarrow (b), (b) \Rightarrow (c), (c) \Rightarrow (d), (d) \Rightarrow (a).

[(a) \Rightarrow (b)] Nech S je strom. To znamená, že S je jednoduchý bezslučkový graf, v ktorom pre každé dva vrcholy u, v z S existuje jediná cesta z u do v . Z toho ale vyplýva, že S je súvislý graf. Stačí už len dokázať, že S je acyklický. To dokážeme nepriamo. Predpokladajme, že v grafe S existuje cyklus z vrcholu u do u . Potom v ňom existuje aj jednoduchý cyklus z u do u . Jeho dĺžka je aspoň 2 (graf S nemá slučky). Preto sa v tomto cykle nachádza vrchol v , ktorý je rôzny od vrcholu u . Zistili sme, že v jednoduchom cykle dĺžky aspoň 2 sa nachádzajú dva rôzne vrcholy u a v . Potom však z vrcholu u do vrcholu v existujú dve rôzne cesty, čo je v spore s tým, že graf S je strom. Preto graf S musí byť acyklický.

[(b) \Rightarrow (c)] Dôkaz urobíme matematickou indukciou vzhľadom na počet n vrcholov grafu S .

Pre $n = 1$ je implikácia evidentne pravdivá - súvislý acyklický graf s jedným vrcholom nemôže obsahovať žiadnu hranu.

Ukážeme, že z pravdivosti tejto implikácie pre $n = k$ vyplýva jej pravdivosť pre $n = k + 1$: Nech teda graf S je súvislý, acyklický a má $k + 1$ vrcholov ($k \geq 1$). Chceme ukázať, že tento graf má k hrán. Vyberme v tomto grafe ťah t maximálnej dĺžky. Ťah t nie je cyklus, lebo graf S je acyklický. Označme u jeden z krajných vrcholov ťahu t . Stupeň vrcholu u musí byť 1, lebo inak by t nebol ťah maximálnej dĺžky v S . Utvoríme podgraf S' grafu S tak, že z S vyhodíme vrchol u a hranu, s ktorou vrchol u inciduje. Graf S' má teda k vrcholov a o jednu hranu menej ako graf S . Navyše graf S' je súvislý a acyklický, preto podľa indukčného predpokladu má $k - 1$ hrán. Graf S má o jednu hranu viac, teda má k hrán.

[(c) \Rightarrow (d)] Nech teda graf S je súvislý a má $n - 1$ hrán. Máme ukázať, že S je acyklický graf. Predpokladajme, že S obsahuje aspoň jeden cyklus. Ak z cyklu odstránime jednu hranu, získame podgraf grafu S , ktorý je súvislý. Ak tento podgraf obsahuje cykly, odstráňme opäť hranu z niektorého cyklu. Takto môžeme pokračovať, až kým nedostaneme graf G^* , ktorý je acyklický. Graf G^* je teda acyklický, súvislý a má n vrcholov. Na základe už dokázanej implikácie (b) \Rightarrow (c) vyplýva, že G^* má $n - 1$ hrán. Potom však graf S má aspoň n hrán, čo je spor. Preto graf S je acyklický.

[(d) \Rightarrow (a)] Predpokladajme, že graf S je acyklický a má $n - 1$ hrán. Musíme ukázať, že S je strom.

Graf S nemôže obsahovať žiadnu slučku, pretože ona s krajným vrcholom tvorí cyklus a S je acyklický. Podobne S nemôže obsahovať viacnásobné hrany. Ak by totiž dve rôzne hrany e_1 a e_2 incidovali s tými istými vrcholmi v a w , tak graf S by obsahoval cyklus (v, e_1, w, e_2, v) . Dokážeme, že S je súvislý (nepriamo). Predpokladajme, že S je nesúvislý a že má k komponentov súvislosti G_1, G_2, \dots, G_k , pričom $k > 1$. Nech G_i má n_i vrcholov, $i \in \{1, \dots, k\}$. Potom platí $n = n_1 + n_2 + \dots + n_k$. Každý komponent súvislosti G_i je súvislý acyklický graf a podľa už dokázaného tvrdenia má $n_i - 1$ hrán a pre počet hrán h grafu S platí

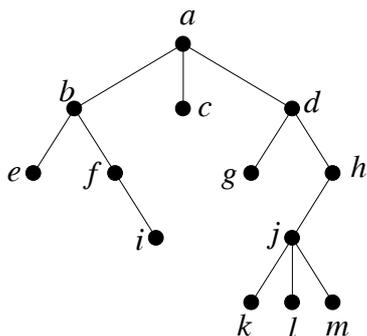
$$h = (n_1 - 1) + (n_2 - 1) + \dots + (n_k - 1) = n_1 + n_2 + \dots + n_k - k = n - k < n - 1,$$

čo je spor. Preto S je súvislý, a teda pre každé dva vrcholy u, v z S existuje aspoň jedna cesta z u do v . Ukážeme ešte, že takáto cesta existuje práve jedna. Ak by existovali aspoň dve rôzne cesty z u do v , tak by v grafe S existoval cyklus, čo nie je možné, lebo graf S je acyklický. \square

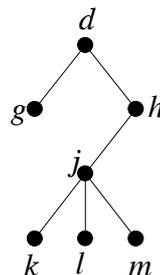
Nech S je strom s koreňom v_0 ; x, y sú vrcholy v S a (v_0, v_1, \dots, v_n) je cesta v S .

Potom

- (a) v_{n-1} sa nazýva **otec** (alebo **rodič**) vrcholu v_n .
- (b) v_0, v_1, \dots, v_{n-1} sú **predkovia** vrcholu v_n .
- (c) v_n sa nazýva **syn** (alebo **dieta**) vrcholu v_{n-1} .
- (d) Ak x je predok y , tak y je **potomok** x .
- (e) Vrchol, ktorý nemá synov, sa nazýva **koncový vrchol**.
- (f) Vrchol, ktorý nie je koncový sa nazýva **vnútorný**.
- (g) **Podstrom s koreňom** x stromu S je jeho podgraf s množinou vrcholov $V = \{x\} \cup \{y; y \text{ je potomok } x\}$ a množinou hrán $E = \{e; e \text{ je hrana cesty z vrcholu } x \text{ do niektorého vrcholu } z \in V\}$.



Obr. 62



Obr. 63

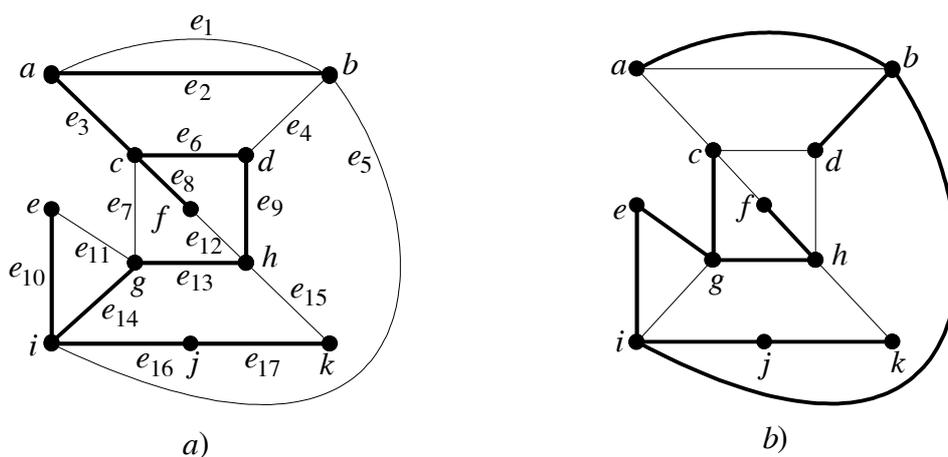
Príklad. V koreňovom strome na obr. 62

- (a) d je otec vrcholu h aj vrcholu g .
- (b) Predkovia vrcholu i sú a, b, f .
- (c) Synovia vrcholu a sú b, c, d .
- (d) Potomkovia vrcholu d sú vrcholy g, h, j, k, l, m .
- (e) Koncové vrcholy sú c, e, g, i, k, l, m .
- (f) Vnútorne vrcholy sú a, b, d, f, h, j .
- (g) Podstrom s koreňom d je na obr. 63.

Kostrou grafu G nazývame každý faktor grafu G , ktorý je stromom.

Inými slovami, kostra grafu G je podgraf grafu G , ktorý je stromom a ktorý obsahuje všetky vrcholy grafu G .

Príklad. Na obr. 64a, 64b sú hrubšími čiarami vyznačené dve rôzne kostry toho istého grafu.



Obr. 64

Veta. Graf G má kostru práve vtedy, keď G je súvislý graf.

Dôkaz. Nech G má kostru T a nech u, v sú vrcholy v G . Keďže T je kostra grafu G , sú u a v vrcholy v T a existuje cesta z u do v . To však znamená, že G je súvislý graf.

Predpokladajme, že G je súvislý graf. Ak G je acyklický graf, tak G je strom. V tomto prípade kostrou grafu G je samotný graf G . V prípade, že v G existuje cyklus, môžeme z tohto cyklu odstrániť jednu hranu, čím vznikne podgraf, ktorý je súvislý a obsahuje tie isté vrcholy ako G . Ak tento podgraf obsahuje cyklus, opäť odstránime z cyklu jednu hranu. Po konečnom počte takýchto krokov získame podgraf T grafu G , ktorý je acyklický a súvislý, čiže strom. A keďže T obsahuje všetky vrcholy grafu G , je T kostrou grafu G . \square

Zdá sa, že dôkaz predchádzajúcej vety poskytuje vhodný algoritmus na konštrukciu kostry grafu. Nie je to tak, lebo vyhľadávanie cyklov v grafe je časovo veľmi náročné. Uvedieme dva vhodnejšie algoritmy.

Algoritmus na vyhľadanie kostry grafu (prehľadávanie do šírky)

Tento algoritmus vytvorí kostru grafu ako koreňový strom, pričom metóda prehľadávania do šírky tkvie v tom, že sa vyhľadávajú najprv všetky vrcholy danej úrovne a až potom sa prejde na nasledujúcu úroveň.

V tomto algoritme P označuje postupnosť vrcholov.

Vstup: Súvislý graf G so zvoleným usporiadaním vrcholov v_1, v_2, \dots, v_n .

Výstup: Kostra T grafu G .

1. [Inicializácia] P obsahuje iba vrchol v_1 . T je graf, ktorý má jeden vrchol v_1 , ten označme ako koreň, a žiadnu hranu.
2. [Pridanie hrán] Pre každý vrchol $x \in P$ (prehľadávanie vrcholov uskutočňujeme podľa zvoleného usporiadania) pridaj do T hranu (x, y) a vrchol y pre všetky $y \in G$ (vo zvolenom poradí) pokiaľ ich pridaním do T nevznikne cyklus. Ak nie je možné pridať do T žiadnu hranu z G , bez toho aby vznikol cyklus, algoritmus končí (T je už kostra grafu G).
3. [Obnova P] Každý vrchol v P nahraď jeho synmi v T a získané vrcholy zorad' podľa na začiatku zvoleného usporiadania. Choď na bod 2.

Veta. Uvedený algoritmus na vyhľadanie kostry grafu je korektný, t.j. po ukončení algoritmu je T kostra grafu G .

Dôkaz. Uvedeným algoritmom vzniká kostra T grafu G postupným pridávaním hrán a vrcholov. Ukážeme, že v každej fáze je T strom a na konci algoritmu T obsahuje všetky vrcholy grafu G . Z toho už vyplýva, že výsledný graf T je kostra grafu G .

Na začiatku T obsahuje jeden vrchol a žiadnu hranu, preto T je strom. Ďalej sa T tvorí vždy pridaním jednej hrany a jedného vrcholu tak, aby nevznikol cyklus. Teda T je vždy acyklický graf a počet jeho vrcholov je o 1 väčší ako počet jeho hrán. To ale znamená, že T je strom. Stačí už len dokázať, že T obsahuje všetky vrcholy grafu G . Dokážeme to nepriamo. Predpokladajme, že strom T po skončení algoritmu neobsahuje všetky vrcholy grafu G . Nech napríklad vrchol x grafu G nie je vrcholom grafu T . Ak by v grafe G existovala cesta z vrcholu x do vrcholu v_1 , povedzme $(x = u_1, h_1, u_2, \dots, h_{k-1}, u_k = v_1)$, tak označme s najmenšie z čísel $i \in \{1, \dots, k\}$ také, že u_i je vrchol v T . Zrejme $1 < s \leq k$. Potom by sme mohli k T pridať hranu h_{s-1} a vrchol u_{s-1} a nevznikol by pritom cyklus, čo je v spore s tým, že algoritmus je

skončený. Teda neexistuje cesta z vrcholu x do vrcholu v_1 . Preto graf G nie je súvislý. To je spor a teda T musí obsahovať všetky vrcholy grafu G . 🔔

Algoritmus na vyhľadanie kostry grafu (prehľadávanie do hĺbky)

Vstup: Súvislý graf G so zvoleným usporiadaním vrcholov v_1, v_2, \dots, v_n .

Výstup: Kostra T grafu G .

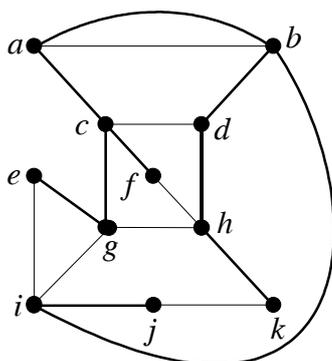
1. [Inicializácia] Polož $w := v_1$. Nech T je graf s jedným vrcholom v_1 a žiadnou hranou. Označme vrchol v_1 ako koreň grafu T .
2. [Pridanie hrany] Vyber hranu (w, v_k) , ktorá nie je v T , s minimálnym k tak, že pridaním hrany (w, v_k) do T nevznikne cyklus. Ak taká hrana neexistuje, choď na 3; inak pridaj hranu (w, v_k) a vrchol v_k do T . Polož $w := v_k$ a choď na 2.
3. [Hotovo?] Ak $w = v_1$, je koniec algoritmu (T je kostra grafu G).
4. [Krok späť] Nech x je otcom w (v strome T). Polož $w := x$ a choď na 2.

Dôkaz korektnosti algoritmu je podobný ako v predošlom algoritme.

Príklad. Prehľadávaním do šírky nájdite kostru grafu G z obr. 64a pri abecednom usporiadaní vrcholov.

Riešenie.

0. $P = (a)$; T obsahuje iba vrchol a .
1. Do T postupne pridáme hrany a vrcholy (jedným krajným vrcholom pridanej hrany musí byť vrchol z P , teda v tomto prípade vrchol a): e_1 alebo e_2, b (nejednoznačnosť výberu hrany je zapríčinená viacnásobnosťou hrany; zvolme napr. hranu e_1); e_3, c . Viac hrán s krajným vrcholom a nie je, preto pokračujeme zmenou postupnosti P : $P = (b, c)$.



Obr. 65

2. Do T postupne pridávame hrany, ktoré majú jeden krajný vrchol v P , a tiež druhé krajné vrcholy týchto hrán (dodržujeme abecedné poradie vrcholov) tak, aby nevznikol cyklus: e_4, d ; e_5, i ; e_8, f ; e_7, g . Utvoríme novú postupnosť synov vrcholov b, c dodržujúc abecedné poradie: $P = (d, f, g, i)$.
3. Pridáme tieto hrany a vrcholy: e_9, h ; e_{11}, e ; e_{16}, j . Nová postupnosť: $P = (e, h, j)$.
4. Pridáme hranu e_{15} a vrchol k . Viac hrán už nie je možné pridať do T bez toho, aby vznikol cyklus (T už obsahuje všetky vrcholy grafu G). Algoritmus je teda ukončený. Nájdená kostra T je na obr. 65 vyznačená hrubšími čiarami. ■

Cvičenia

1. Existuje strom, ktorého všetky vrcholy majú stupeň 2?
2. V algoritmoch na nájdenie kostry grafu sa testuje, či pridaním hrany do T vznikne cyklus. Dokážte, že ak x je vrchol v T a (x, y) je hrana v G ale nie v T , tak pridaním hrany (x, y) do T vznikne cyklus v T práve vtedy, keď y je vrchol v T .
3. Nájdiť kostru grafu G z obr. 64a prehľadávaním do šírky pri usporiadaní vrcholov $k, j, i, h, g, f, e, d, c, b, a$.

Ohodnotené grafy

[Orientovaný] graf sa nazýva **ohodnotený**, ak každej jeho hrane h je priradené reálne číslo $w(h)$, ktoré nazývame **ohodnotenie hrany h** . V diagrame [orientovaného] grafu sa ohodnotenie hrany pripisuje k príslušnej hrane.

Príkladom ohodnoteného grafu je napr. cestná sieť určitého regiónu, kde obce tohto regiónu sú vrcholy a cesty spájajúce tieto cesty sú hrany tohto grafu. Ohodnotením hrany môže byť dĺžka tejto cesty alebo finančné náklady na jej výstavbu.

Dĺžkou [orientovaného] sledu S v ohodnotenom [orientovanom] grafe G nazývame súčet ohodnotení všetkých [orientovaných] hrán [orientovaného] sledu S . V prípade, že [orientovaný] sled S neobsahuje žiadnu hranu, nazveme jeho dĺžkou číslo 0. Dĺžku [orientovaného] sledu S budeme označovať $d(S)$.

[Orientovaný] ťah z vrcholu u do vrcholu v , ktorý má zo všetkých ťahov z u do v najmenšiu dĺžku, sa nazýva **najkratší [orientovaný] ťah z vrcholu u do vrcholu v** . Jeho dĺžka sa nazýva **vzdialenosť vrcholu v od vrcholu u** v ohodnotenom [orientovanom] grafe G a budeme ju označovať $d(u, v)$. Ak neexistuje [orientovaný] sled z u do v , tak definujeme $d(u, v) = \infty$.

V tejto časti budeme pracovať s rozšírenou množinou reálnych čísel $\mathbf{R} = \mathbf{R} \cup \{\infty\}$. Usporiadanie reálnych čísel a operáciu sčítavania reálnych čísel rozšírime na množinu \mathbf{R} takto: pre všetky $a \in \mathbf{R}$, $b \in \mathbf{R}$ definujeme $a < \infty$, $b + \infty = \infty + b = \infty$.

Veta. *Nech G je ohodnotený [orientovaný] graf s nezápornými ohodnoteniami hrán, u, v sú vrcholy v G a S je sled z u do v . Potom $d(u, v) \leq d(S)$.*

Dôkaz. Odstránením cyklov zo sledu S získame ťah S' z vrcholu u do vrcholu v . Vzhľadom na nezápornosť ohodnotení hrán potom platí $d(S') \leq d(S)$. Odtiaľ dostávame $d(u, v) \leq d(S') \leq d(S)$, lebo $d(u, v)$ je dĺžka najkratšieho ťahu z u do v . 🔔

Veta. *Nech G je ohodnotený graf s nezápornými ohodnoteniami hrán. Potom pre všetky vrcholy u, v, w ohodnoteného grafu G platí:*

- (a) $d(u, v) = d(v, u)$,
 (b) $d(u, v) + d(v, w) \geq d(u, w)$.

Dôkaz.

(a) Pre ľubovoľný ťah $S = (u, h_1, u_1, \dots, u_{n-1}, h_n, v)$ z vrcholu u do vrcholu v je $S' = (v, h_n, u_{n-1}, \dots, u_1, h_1, u)$ ťah z vrcholu v do vrcholu u a navyše $d(S) = d(S')$. Z toho potom vyplýva, že ak S je najkratší ťah z u do v , tak S' je najkratší ťah z v do u a

$$d(u, v) = d(S) = d(S') = d(v, u).$$

(b) Nech

$$S_1 = (u, h_1, u_1, \dots, u_{n-1}, h_n, v), S_2 = (v, e_1, v_1, \dots, v_{m-1}, e_m, w)$$

sú najkratšie ťahy, prvý z vrcholu u do vrcholu v a druhý z vrcholu v do vrcholu w . Potom

$$S = (u, h_1, u_1, \dots, u_{n-1}, h_n, v, e_1, v_1, \dots, v_{m-1}, e_m, w)$$

je sled z vrcholu u do vrcholu w a pre jeho dĺžku evidentne platí $d(S) = d(u, v) + d(v, w)$. Pre dĺžku $d(u, w)$ najkratšieho ťahu z u do w potom zrejme platí

$$d(u, w) \leq d(S) = d(u, v) + d(v, w). \quad \text{🔒}$$

Poznámka. Časť (b) predchádzajúcej vety platí aj pre orientované grafy. Časť (a) však pre ne neplatí.

Veta. Nech $S = (u_0, h_1, u_1, \dots, u_{n-1}, h_n, u_n)$ je najkratší [orientovaný] ťah z vrcholu u_0 do vrcholu u_n v ohodnotenom [orientovanom] grafe G s nezáporne ohodnotenými hranami. Potom pre všetky $i, j \in \{0, 1, \dots, n\}$, $i \neq j$ je $S_2 = (u_i, h_{i+1}, \dots, h_j, u_j)$ najkratší [orientovaný] ťah z vrcholu u_i do vrcholu u_j .

Dôkaz. Nepriamo. Predpokladajme, že $S_2 = (u_i, h_{i+1}, u_{i+1}, \dots, u_{j-1}, h_j, u_j)$ nie je najkratší [orientovaný] ťah z u_i do u_j ale je ním $S'_2 = (u_i, e_1, v_1, \dots, v_{j-i-1}, e_{j-i-1}, u_j)$, a teda platí

$$d(u_i, u_j) = d(S'_2) < d(S_2).$$

Označme

$$S_1 = (u_0, h_1, u_1, \dots, u_{i-1}, h_i, u_i), S_3 = (u_j, h_{j+1}, u_{j+1}, \dots, u_{n-1}, h_n, u_n).$$

Potom pre sled

$$S' = (u_0, h_1, u_1, \dots, u_{i-1}, h_i, u_i, e_1, v_1, \dots, v_{j-i-1}, e_{j-i-1}, u_j, h_{j+1}, u_{j+1}, \dots, u_{n-1}, h_n, u_n)$$

z vrcholu u_0 do vrcholu u_n platí

$$d(S') = d(S_1) + d(S'_2) + d(S_3) < d(S_1) + d(S_2) + d(S_3) = d(S).$$

Potom však $d(u_0, u_n) \leq d(S') < d(S) = d(u_0, u_n)$, čo je spor. 🔒

Uvedieme teraz Dijkstrov algoritmus na výpočet vzdialeností vrcholov v ohodnotenom grafe G , v ktorom ohodnotenia všetkých hrán sú nezáporne. Pre zjednodušenie, aby bola zaručená existencia najkratších sledov z ľubovoľného vrcholu do ľubovoľného vrcholu, budeme predpokladať, že G je súvislý graf.

Dijkstrov algoritmus určí vzdialenosť každého vrcholu od pevne zvoleného vrcholu b . Základom tohto algoritmu je priradenie značky každému vrcholu v grafu G . Táto značka môže byť dočasná alebo trvalá. V oboch prípadoch je to dvojica $(L(v), P(v))$, kde $L(v)$ je nezáporné reálne číslo a $P(v)$ je vrchol grafu G . Každý vrchol má na začiatku dočasnú značku, ktorá sa môže meniť. Dočasná značka sa neskôr stáva trvalou a odvtedy sa už nemení. Množinu všetkých vrcholov grafu G s dočasnou značkou budeme označovať D . Zložka $L(v)$ určuje dĺžku toho ťahu z b do v , ktorý má z dovtedy prehľadaných ťahov z b do v najmenšiu dĺžku. Predposledný vrchol tohto ťahu je $P(v)$. Neskôr ukážeme, že akonáhle značka vrcholu v je trvalá, tak $L(v)$ je dĺžka najkratšieho ťahu z b do v , čiže je to vzdialenosť vrcholu v od vrcholu b .

Poznamenajme ešte, že pri hľadaní najkratších ťahov v ohodnotených multigrafoch stačí pri viacnásobnej hrane s krajnými vrcholmi u, v uvažovať iba hranu, incidentnú s týmito vrcholmi, ktorá má najmenšie ohodnotenie. Preto sa stačí obmedziť iba na jednoduché grafy.

Dijkstrov algoritmus

Tento algoritmus určí dĺžku najkratšieho ťahu z vybraného vrcholu b do každého vrcholu v (čiže $d(b, v)$) ohodnoteného jednoduchého súvislého grafu G . Ohodnotenie hrany (x, y) je $w(x, y) \geq 0$. Značka každého vrcholu x je $(L(x), P(x))$, pričom na konci algoritmu $L(x)$ je dĺžka najkratšieho ťahu z b do x a $P(x)$ je predposledný vrchol tohto ťahu.

Vstup: Ohodnotený súvislý jednoduchý graf $G = (V, E, e)$; vybraný vrchol $b \in V$.

Výstup: Trvalá značka $(L(v), P(v))$ pre každý vrchol $v \in V$.

1. [Inicializácia] $L(b) := 0, P(b) := b$. Pre všetky $x \in V - \{b\}$ polož $L(x) := \infty, P(x) := b$.
2. [Koniec?] Ak $D = \emptyset$, tak skonči.
3. [Vytvorenie trvalej značky] Vyber vrchol $u \in D$ s najmenšou hodnotou $L(u)$ a polož $D := D - \{u\}$.
4. [Zmena dočasných značiek] Pre každý vrchol $x \in D$ susedný s vrcholom u vykonaj: ak $L(x) > L(u) + w(u, x)$, tak $L(x) := L(u) + w(u, x), P(x) := u$. Choď na krok 2.

Veta. Pre každý vrchol v ohodnoteného súvislého jednoduchého grafu G platí: ak $(L(v), P(v))$ je trvalá značka vrcholu v , vytvorená Dijkstrovym algoritmom, tak $L(v)$ je dĺžka najkratšieho ťahu z b do v a pre $v \neq b$ je $P(v)$ predposledný vrchol tohto ťahu.

Dôkaz. Uskutočníme ho matematickou indukciou vzhľadom na počet vykonaných iterácií, t.j. počet vykonaných cyklov 2-4 Dijkstrovho algoritmu. Najprv dokážeme tvrdenia:

A. Ak v značke vrcholu $v, v \neq b$, je $L(v) < \infty$, tak $L(v)$ je dĺžka niektorého ťahu z vrcholu b do vrcholu v a $P(v)$ je predposledný vrchol tohto ťahu.

B. Nech po m -tej iterácii má vrchol x trvalú značku a vrchol y dočasnú značku. Potom $L(x) \leq L(y)$.

Dôkaz tvrdenia A: Až v druhej iterácii vzniknú značky vrcholov $v, v \neq b$, v ktorých je $L(v) < \infty$. Týmito vrcholmi sú všetky susedné vrcholy s vrcholom b . Pre každý takýto vrchol v je podľa kroku 4 Dijkstrovho algoritmu $L(v) = L(b) + w(b, v) = w(b, v)$, čo je dĺžka ťahu (b, v) a $P(v) = b$, čo je predposledný vrchol tohto ťahu.

Predpokladajme, že tvrdenie platí pre všetky značky v $(m-1)$ -tej iterácii. Nech značka vrcholu v sa v m -tej iterácii zmení (ak by sa nezmenila, tak je rovnaká ako v $(m-1)$ -tej

iterácii a podľa indukčného predpokladu tvrdenie pre ňu platí). Potom platí $L(v) = L(u) + w(u, v)$, kde u je vrchol, ktorého značka sa v m -tej iterácii zmenila na trvalú, vrcholy u, v sú zrejme susedné a $P(v) = u$. Trvalá značka vrcholu u je presne tá istá ako na konci $(m - 1)$ -tej iterácie, preto podľa indukčného predpokladu je $L(u)$ dĺžka niektorého ťahu z b do u . Povedzme, že tým ťahom je (b, u_1, \dots, u_k, u) . Potom $(b, u_1, \dots, u_k, u, v)$ je ťah z b do v , pričom jeho dĺžka je $L(v)$ a predposledný vrchol v tomto ťahu je $P(v)$.

Dôkaz tvrdenia **B**: Po prvej iterácii má trvalú značku jedine vrchol b . Preto $x = b$ a $L(x) = L(b) = 0$. Ak y a b nie sú susedné vrcholy, tak $L(y) = \infty$. Ak y a b sú susedné vrcholy, tak $L(y) = L(b) + w(b, y) = w(b, y) \geq 0$. V oboch prípadoch je $L(y) \geq L(x)$.

Predpokladajme, že tvrdenie platí pre všetky značky vytvorené do $(m - 1)$ -tej iterácie. Pre vrchol y môžu nastať dva prípady:

- a) jeho dočasná značka v m -tej iterácii je rovnaká ako v $(m - 1)$ -tej iterácii,
- b) jeho dočasné značky v m -tej a $(m - 1)$ -tej iterácii sú rôzne.

Upresníme ešte, že výraz „dočasná značka vrcholu v k -tej iterácii“ znamená „dočasná značka vrcholu po vykonaní kroku 4 v k -tej iterácii Dijkstrovho algoritmu“. Ak nastane prípad a), tak podľa indukčného predpokladu je $L(y) \geq L(x)$ pre všetky vrcholy x , ktorých trvalá značka bola vytvorená najneskôr v $(m - 1)$ -tej iterácii. V prípade, že trvalá značka vrcholu x vznikla v m -tej iterácii, tak podľa kroku 3 Dijkstrovho algoritmu musí platiť $L(x) \leq L(y)$, v opačnom prípade by totiž v m -tej iterácii nemohlo dôjsť k zmene dočasnej značky vrcholu x na trvalú. Ak pre vrchol y nastane prípad b), tak to znamená, že v m -tej iterácii došlo k zmene dočasnej značky vrcholu y a pre prvú zložku novej značky platí $L(y) = L(x) + w(x, y) \geq L(x)$.

Teraz už môžeme pristúpiť k dôkazu vety, pričom vzhľadom na tvrdenie **A** stačí dokázať len časť vety týkajúcej sa prvej zložky trvalej značky. V prvej iterácii má trvalú značku len vrchol b , pričom $L(b) = 0 = d(b, b)$.

Predpokladajme, že tvrdenie vety platí pre všetky vrcholy, ktorých trvalé značky boli vytvorené najneskôr v $(m - 1)$ -tej iterácii. Nech v m -tej iterácii bola trvalá značka priradená vrcholu v a nech $S = (b = v_0, v_1, \dots, v_{k-1}, v_k = v)$ je najkratší ťah z vrcholu b do vrcholu v . Pre vrcholy $v_0, v_1, \dots, v_{k-1}, v_k$ platí: buď aspoň jeden z nich má dočasnú značku, alebo všetky majú trvalú značku.

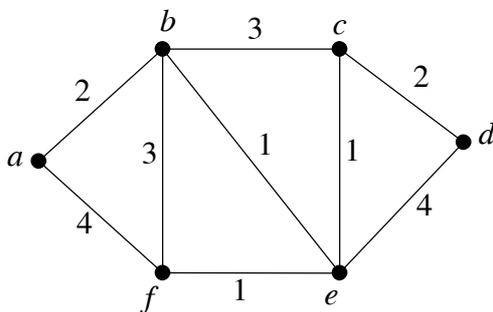
Predpokladajme, že nastal prvý prípad. Vyberme z týchto vrcholov s dočasnou značkou ten s najmenším indexom. Nech je to vrchol v_i . Keďže vrcholy v_0 a v_k majú trvalú značku je $i \in \{1, \dots, k - 1\}$. Vrchol v_{i-1} má trvalú značku a podľa indukčného predpokladu je $L(v_{i-1})$ dĺžka najkratšieho ťahu z b do v_{i-1} , ktorým je napr. ťah $(b = v_0, v_1, \dots, v_{i-1})$. Pre dĺžku ťahu S zrejme platí

$$d(S) \geq L(v_{i-1}) + w(v_{i-1}, v_i) \geq L(v_i) \geq L(v).$$

Predposledná nerovnosť vyplýva z vlastnosti dočasných značiek uvedenej v kroku 4 Dijkstrovho algoritmu a posledná nerovnosť vyplýva z tvrdenia **B**. Na druhej strane, vzhľadom na to, že $L(v)$ je dĺžka niektorého a $d(S)$ najkratšieho ťahu z b do v , musí platiť $L(v) \geq d(S)$, čo s predošlou nerovnosťou dáva $L(v) = d(S)$.

Uvažujme teraz o prípade, keď všetky vrcholy ťahu S majú trvalú značku. Potom podľa indukčného predpokladu je $L(v_{k-1})$ dĺžka najkratšieho ťahu, napr. (b, v_1, \dots, v_{k-1}) , z b do v_{k-1} . Pre dĺžku ťahu S platí $d(S) = L(v_{k-1}) + w(v_{k-1}, v)$. Vzhľadom na to, že dočasná značka vrcholu v na konci $(m - 1)$ -tej iterácie je rovnaká ako jeho trvalá značka, musí podľa kroku 4 Dijkstrovho algoritmu platiť $L(v) \leq L(v_{k-1}) + w(v_{k-1}, v) = d(S)$. Z druhej strany $L(v)$ je dĺžkou niektorého ťahu z b do v a preto podľa tvrdenia **B** je $L(v) \geq d(S)$. Z posledných dvoch nerovností vyplýva $L(v) = d(S)$. ☺

Príklad. Nájdiť vzdialenosti vrcholu a od ostatných vrcholov a najkratší ťah z vrcholu a do vrcholu d v ohodnotenom grafe G z obr. 66.



Obr. 66

Riešenie úlohy uskutočníme pomocou Dijkstrovho algoritmu.

0. Určíme počiatkové hodnoty dočasných značiek všetkých vrcholov grafu G : $a \dots (0, a)$, $b \dots (\infty, b)$, $c \dots (\infty, c)$, $d \dots (\infty, d)$, $e \dots (\infty, e)$, $f \dots (\infty, f)$. Množina dočasných značiek je $D = \{a, b, c, d, e, f\}$.
1. Najmenšiu hodnotu prvej zložky dočasnej značky má vrchol a , preto podľa kroku 3 Dijkstrovho algoritmu sa jeho značka stáva trvalou a množina vrcholov s dočasnou značkou je $D := \{b, c, d, e, f\}$. Určíme teraz nové hodnoty dočasných značiek susedných vrcholov b, f vrcholu a . Keďže platí:

$$L(a) + w(a, b) = 0 + 2 = 2 < \infty = L(b),$$

$$L(a) + w(a, f) = 0 + 4 = 4 < \infty = L(f),$$

zmenia sa dočasné značky vrcholov b, f na $(2, a)$, $(4, a)$ v uvedenom poradí.

Doterajší výpočet môžeme stručne zapísať do tabuľky, v ktorej na označenie trvalej značky sme použili rámček:

	a	b	c	d	e	f
0.	$(0, a)$	(∞, b)	(∞, c)	(∞, d)	(∞, e)	(∞, f)
1.	$\boxed{(0, a)}$	$(2, a)$	-	-	-	$(4, a)$

2. Z dočasných značiek najmenšiu hodnotu prvej zložky má vrchol b . Jeho značka sa stáva trvalou, $D := \{c, d, e, f\}$. Dočasné značky sa môžu zmeniť iba pri vrcholoch c, e, f , ktoré sú susedné k vrcholu b . Vzhľadom k tomu, že

$$L(b) + w(b, c) = 2 + 3 = 5 < \infty = L(c),$$

$$L(b) + w(b, e) = 2 + 1 = 3 < \infty = L(e),$$

$$L(b) + w(b, f) = 2 + 3 = 5 > 4 = L(f),$$

zmenia sa dočasné značky iba vrcholov c, e na $(5, b)$, $(3, b)$ v uvedenom poradí (obr. 67).

3. Značka vrcholu e sa stáva trvalou, $D := \{c, d, f\}$. Zmena dočasných značiek:

$$L(e) + w(e, c) = 3 + 1 = 4 < 5 = L(c), \quad L(c) := 4, \quad P(c) := e,$$

$$L(e) + w(e, d) = 3 + 4 = 7 < \infty = L(d), \quad L(d) := 7, \quad P(d) := e,$$

$$L(e) + w(e, f) = 3 + 1 = 4 = L(f)$$

4. Najmenšiu prvú zložku dočasnej značky majú teraz vrcholy c a f . Lubovoľnú (ale len jednu) z týchto dvoch dočasných značiek môžeme označiť ako trvalú. Nech je to značka

vrcholu c , teda $D := \{d, f\}$. Zmena dočasných značiek:

$$L(c) + w(c, d) = 4 + 2 = 6 < 7 = L(d), \quad L(d) := 6, \quad P(d) := c.$$

5. Vrchol f získava trvalú značku. $D := \{d\}$. Dočasná značka vrcholu d sa nemení (vrcholy d, f nie sú susedné).
6. Značka vrcholu d sa stáva trvalou, $D := \emptyset$, algoritmus končí.

Vzdialenosti vrcholov grafu G od vrcholu a sú v prvej zložke ich trvalých značiek (obr. 67), teda $d(a, a) = 0$, $d(a, b) = 2$, $d(a, c) = 4$, $d(a, d) = 6$, $d(a, e) = 3$, $d(a, f) = 4$.

Najkratší ťah S z vrcholu a do vrcholu d určíme z druhých zložiek trvalých značiek. V trvalej značke vrcholu d je $P(d) = c$, čo je predposledný vrchol ťahu S . Keďže najkratší ťah z vrcholu a do vrcholu c je časťou najkratšieho ťahu S z vrcholu a do d , je $P(c) = e$ ďalší vrchol ťahu S bezprostredne predchádzajúci vrcholu c . Ďalšie vrcholy ťahu predchádzajúce vrcholu e sú v poradí: $P(e) = b$, $P(b) = a$. Najkratší ťah z a do d je teda sled $S = (a, b, e, c, d)$. ■

	a	b	c	d	e	f
0.	$(0, a)$	(∞, b)	(∞, c)	(∞, d)	(∞, e)	(∞, f)
1.	$(0, a)$	$(2, a)$	-	-	-	$(4, a)$
2.		$(2, a)$	$(5, b)$	-	$(3, b)$	-
3.			-	$(7, e)$	$(3, b)$	-
4.			$(4, b)$	$(6, c)$		-
5.				-		$(4, a)$
6.				$(6, c)$		

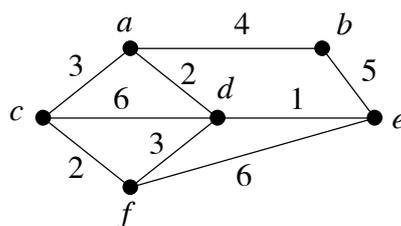
Obr. 67

Dijkstrov algoritmus je možné použiť aj na určenie vzdialeností vrcholov a najkratších orientovaných ťahov pre ohodnotené orientované grafy. K tomu stačí v 4. kroku tohto algoritmu text „Pre každý vrchol $x \in D$ susedný s vrcholom u “ nahradiť textom „Pre každý vrchol $x \in D$, ktorý je koncovým vrcholom hrany so začiatočným vrcholom u “.

Príklad. Domy sídliska je potrebné prepojiť káblom tak, aby signál vedený v kábli sa dostal do každého domu a aby cena tohto káblového rozvodu bola minimálna. Jednu schému domov a ich možného prepojenia znázorňuje graf G na obr. 68. Ohodnotenia hrán sú úmerné cene príslušného káblového prepojenia.

Riešenie. Riešením tejto úlohy, povedané jazykom teórie grafov, je podgraf T grafu G s minimálnym súčtom ohodnotení hrán. Z požiadaviek úlohy tiež vyplýva, že T je súvislý graf, ktorý obsahuje všetky vrcholy grafu G a pre všetky vrcholy u, v z grafu G existuje jediná cesta z vrcholu u do vrcholu v . T je teda kostra grafu G s minimálnym súčtom ohodnotení hrán. ■

Nech G je ohodnotený graf. **Minimálna kostra grafu G** je jeho kostra T s minimálnym súčtom ohodnotení všetkých hrán v T .



Obr. 68

Existuje niekoľko algoritmov na vytvorenie minimálnej kostry. Uvedieme jeden z nich.

Primov algoritmus

Vstup: Súvislý ohodnotený graf G s vrcholmi v_1, v_2, \dots, v_n .

Výstup: Minimálna kostra T grafu G .

1. [Inicializácia] T je graf s vrcholom v_1 , bez hrán.
2. [Koniec?] Ak T má $n - 1$ hrán, algoritmus končí.
3. [Pridaj hranu] Medzi všetkými hranami z G , ktoré nie sú v T , ale incidujú s vrcholom v T , a ich pridaním do T nevznikne cyklus, vyber hranu s najmenším ohodnotením. Ak viac ako jedna hrana má minimálne ohodnotenie, vyber z nich hranu (v_i, v_j) s najmenším i , povedzme i_0 . Ak aspoň dve hrany (v_{i_0}, v_j) majú to isté minimálne ohodnotenie, vyber hranu s najmenším j . Vybranú hranu aj s krajným vrcholom pridaj do T . Choď na krok 2.

Veta. Primov algoritmus je korektný, t.j. po jeho ukončení je T minimálna kostra grafu G .

Dôkaz. Priamo z opisu algoritmu vyplýva, že po jeho ukončení je T acyklický podgraf grafu G , ktorý obsahuje $n - 1$ hrán a všetkých n vrcholov grafu G , teda T je kostra grafu G . Všimnime si, že kostra T sa vytvára postupne. Na začiatku obsahuje jeden vrchol a jednu hranu, v každej ďalšej iterácii sa do T pridáva jedna hrana a jeden vrchol. Dôkaz toho, že T je minimálna kostra grafu G , urobíme tak, že matematickou indukciou ukážeme, že po každej iterácii je T podgraf niektorej minimálnej kostry grafu G . Teda po ukončení algoritmu je T podgrafom minimálnej kostry T' grafu G , odkiaľ vzhľadom na rovnaký počet vrcholov a hrán vyplýva, že grafy T a T' sú totožné a teda T je minimálna kostra grafu G .

Nech T_k označuje strom vytvorený pomocou Primovho algoritmu v k -tej iterácii.

Ak $k = 0$, T_0 pozostáva z jedného vrcholu. Zrejme T_0 je podgrafom každej minimálnej kostry grafu G .

Predpokladajme, že T_k je podgrafom niektorej minimálnej kostry T' grafu G . V nasledujúcej iterácii z množiny H_{k+1} všetkých hrán grafu G , ktorých práve jeden krajný vrchol patrí do T_k , vyberieme hranu s najmenším ohodnotením. Nech je to hrana (v_i, v_j) , pričom v_i je a v_j nie je vrcholom grafu T_k . Pridaním hrany (v_i, v_j) a vrcholu v_j do T_k vznikne T_{k+1} . V prípade, že hrana (v_i, v_j) patrí kostre T' , je T_{k+1} podgrafom kostry T' a dôkaz je skončený. Ak tento prípad nenastane, tak v T' existuje cesta z vrcholu v_i do vrcholu v_j a v nej taká hrana (x, y) , že x patrí a y nepatrí do T_k . Keďže hrana $(x, y) \in H_{k+1}$, pre jej ohodnotenie platí $w(v_i, v_j) \leq w(x, y)$. Pridajme hranu (v_i, v_j) do kostry T' . Teraz z kostry T' vytvoríme novú minimálnu kostru T'' grafu G , ktorej podgrafom bude T_{k+1} . Ak k stromu T' pridáme hranu (v_i, v_j) , dostaneme graf, ktorého podgrafom je graf T_{k+1} . Tento nový graf však obsahuje cyklus. Odstránením hrany (x, y) z tohto grafu, získame graf T'' , ktorý je acyklický, má rovnaký počet vrcholov a hrán

ako T' , teda T'' je kostra grafu G a T_{k+1} je podgrafom T'' . Zo vzťahu medzi ohodnoteniami hrán (v_i, v_j) a (x, y) vyplýva, že T'' je minimálna kostra grafu G . 🔔

Príklad. Nájďme minimálnu kostru grafu G z obr. 68.

Riešenie. Zvolíme abecedné poradie vrcholov. Na začiatku graf T je tvorený vrcholom a , nemá žiadne hrany. Vyhľadajme teraz všetky hrany, ktorých jeden krajný vrchol je v T a ich pridaním do T nevznikne cyklus:

1. (a, b) , (a, c) , (a, d) (Číslo uvedené nad hranou udáva ohodnotenie tejto hrany.)

Z týchto hrán vyberieme hranu s minimálnym ohodnotením (je označená podčiarknutím) a pridáme ju aj s druhým krajným vrcholom do T . Túto procedúru opakujeme, až kým T neobsahuje všetky vrcholy grafu G .

2. (a, b) , (a, c) , (d, c) , (d, e) , (d, f)

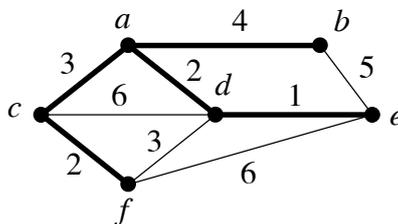
3. (a, b) , (a, c) , (d, c) , (d, f) , (e, b) , (e, f)

Z dvoch hrán (a, c) , (d, f) s rovnakým minimálnym ohodnotením sme vybrali prvú, lebo v abecednom usporiadaní a predchádza d .

4. (a, b) , (c, f) , (d, f) , (e, b) , (e, f)

5. (a, b) , (e, b)

Nájdená minimálna kostra grafu G je na obr. 69 vyznačená hrubšími čiarami. ■



Obr. 69

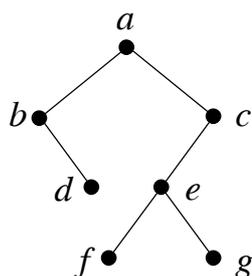
Cvičenia

1. Zadajte ohodnotený orientovaný graf G , pre ktorý nie je pravdivé tvrdenie: pre každé vrcholy u, v z G je $d(u, v) = d(v, u)$.
2. Určte ohodnotený graf, v ktorom existujú vrcholy u, v a sled S z u do v pre ktoré platí $d(u, v) > d(S)$.
3. Uvedte príklad ohodnoteného grafu, v ktorom pre niektoré vrcholy u, v, w platí $d(u, w) > d(u, v) + d(v, w)$.
4. Nech G je ohodnotený orientovaný graf s nezáporne ohodnotenými hranami. Dokážte, že pre všetky jeho vrcholy u, v, w platí $d(u, v) + d(v, w) \geq d(u, w)$.
5. Nech G je ohodnotený [orientovaný] graf s kladnými ohodnoteniami všetkých hrán a nech pre jeho vrcholy u, v existuje [orientovaný] sled z u do v . Dokážte, že najkratší [orientovaný] sled z u do v je [orientovaná] cesta.

Binárne stromy

Binárne stromy sú špeciálnym prípadom koreňových stromov. Ich význam je v ich využití v informatike.

Binárny strom je koreňový strom, v ktorom každý vrchol má najviac dvoch synov. Z nich jeden je označený ako *ľavý syn* (v diagrame sa kreslí vľavo) a druhý ako *pravý syn* (v diagrame sa kreslí vpravo). Ten istý vrchol nemôže byť súčasne ľavým aj pravým synom.



Obr. 70

Príklad. V binárnom strome na obr. 70 je vrchol b ľavým a vrchol c pravým synom vrcholu a . Vrchol d je pravým synom vrcholu b , vrchol b nemá ľavého syna. Vrchol e je ľavým synom vrcholu c , vrchol c nemá pravého syna. ■

Úplný binárny strom je binárny strom, v ktorom každý vrchol buď má dvoch synov alebo nemá žiadneho syna.

Veta. Nech T je úplný binárny strom, ktorý má n vnútorných vrcholov. Potom T má $n + 1$ koncových vrcholov a počet všetkých jeho vrcholov je $2n + 1$.

Dôkaz. Binárny strom T má dva typy vrcholov: synov a vrcholy, ktoré nie sú synmi. V T existuje jeden vrchol, ktorý nie je syn, a je to koreň stromu T . Všetky ostatné vrcholy sú synovia. Strom T má n vnútorných vrcholov, z ktorých každý má 2 synov. Z toho vyplýva, T má $2n$ synov. Potom všetkých vrcholov v T je $2n + 1$ a počet koncových vrcholov je $(2n + 1) - n = n + 1$. 

Jeden zo spôsobov využitia úplných binárnych stromov v informatike je pri kódovaní alfanumerických znakov do postupností núl a jednotiek. Obvykle sa tieto znaky kódujú do nula-jednotkových postupností konštantnej dĺžky, ako napr. ASCII (American Standard Code for Information Interchange). Používajú sa však aj kódy premenlivej dĺžky. Takými sú napr. Huffmanove kódy. Budeme ich definovať pomocou úplných binárnych stromov.

Nech $A = \{a_1, \dots, a_n\}$ je množina znakov, T je ohodnotený úplný binárny strom s n koncovými vrcholmi označenými a_1, \dots, a_n a koreňom u . Ohodnotenie hrany

h stromu T je

$$w(h) = \begin{cases} 1, & \text{ak } h \text{ spája vrchol s jeho ľavým synom} \\ 0, & \text{ak } h \text{ spája vrchol s jeho pravým synom} \end{cases}$$

Nech $(u, h_1, u_1, \dots, u_{k-1}, h_k, a)$ je cesta z koreňa u do vrcholu $a \in A$. Potom postupnosť (reťazec) núl a jednotiek

$$w(h_1) w(h_2) \dots w(h_k)$$

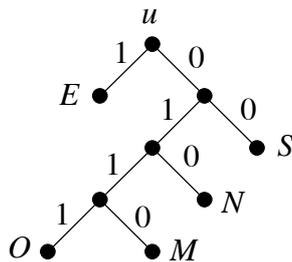
sa nazýva **Huffmanov kód znaku** a .

Príklad. Huffmanov kód pre množinu znakov $\{E, M, N, O, S\}$ je definovaný ohodnoteným binárnym stromom na obr. 71.

- Vypíšme kódy jednotlivých znakov.
- Zakódujme slovo MENO.
- Dekódujme reťazec 01001110001101.

Riešenie.

- E ... 1, M ... 0110, N ... 010, O ... 0111, S ... 00.
- Kód slova vytvoríme tak, že zapíšeme za sebou kódy jednotlivých znakov daného slova: MENO ... 011010100111.



Obr. 71

c) Pri dekódovaní reťazca využívame binárny strom, ktorým je definovaný kód - kódovací strom. Vychádzame z koreňa tohto stromu. Načítame prvý znak reťazca, keďže je to 0, presunúme sa hranou vpravo do nasledujúceho vrcholu. Teraz načítame ďalší znak, je to 1, takže sa presunieme do ďalšieho vrcholu hranou vľavo. Ďalší znak je 0 - presun vľavo. Dostali sme sa do koncového vrcholu N, čo je prvý znak hľadaného slova. V kódovacom strome sa teraz presunieme do koreňa, načítaním ďalších znakov reťazca a následným pohybom v strome vľavo resp. vpravo, až kým sa nedostaneme do niektorého koncového vrcholu, získame ďalší znak hľadaného slova. Takýmto spôsobom určíme nakoniec celé dekódované slovo NOSME. ■

Zakódovaním jedného slova rôznymi Huffmanovými kódmi môžeme získať reťazce núl a jednotiek rôznych dĺžok. Naskytá sa otázka, či je možné zostrojiť optimálny Huffmanov kód, t.j. kód, v ktorom by toto slovo bolo zakódované najkratším reťazcom. Odpoveď na túto otázku je kladná. Optimálny Huffmanov kód má zrejme tú vlastnosť, že znak s menším počtom výskytov v danom slove (menšou frekvenciou výskytu) nemá dlhší kód ako znak s väčšou frekvenciou. Huffman vytvoril algoritmus, ktorý skonštruuje optimálny Huffmanov

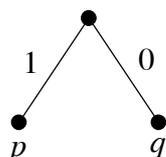
kód pre znaky s požadovanou frekvenciou výskytu. Výstupom tohto algoritmu je ohodnotený binárny strom, ktorého koncové vrcholy sú označené frekvenciou jednotlivých znakov. Z tohto stromu dostaneme kódovací strom prepísaním každej frekvencie znakom, ktorý má túto frekvenciu.

Algoritmus konštrukcie optimálneho Huffmanovho kódu

Vstup: Postupnosť n početností, $n \geq 2$.

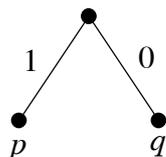
Výstup: Úplný binárny strom T , ktorý určuje optimálny Huffmanov kód.

1. [Základný krok] Ak $n = 2$ a p, q sú frekvencie, tak výstupom je



a skončí.

2. [Rekurzívny krok] Nech p, q sú najmenšie frekvencie. Utvor nový zoznam frekvencií nahradením dvoch frekvencií p, q jednou $p + q$. Zavolaj tento algoritmus pri vstupe s novým zoznamom frekvencií. V jeho výstupe nahraď vrchol označený $p + q$ stromom



a dostaneš výstup T .

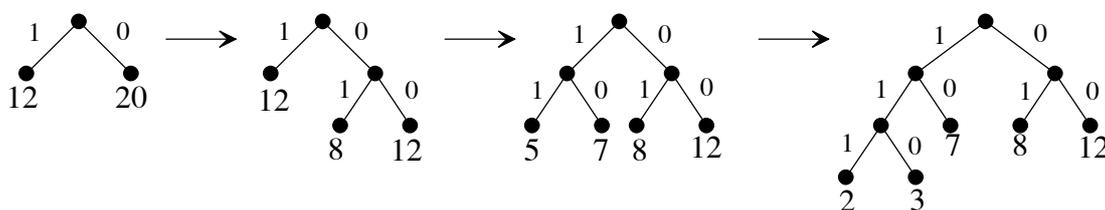
Príklad. Zostrojme optimálny Huffmanov kód pre znaky s požadovanou frekvenciou podľa tabuľky

znak	frekvencia
!	2,00
@	3,00
#	7,00
\$	8,00
%	12,00

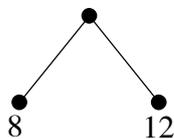
Riešenie. Uvedený algoritmus začína opakovaným nahradením najmenších dvoch frekvencií ich súčtom, až kým nedostaneme dvojčlennú postupnosť frekvencií:

$$\begin{aligned}
 2, 3, 7, 8, 12 &\rightarrow 2+3, 7, 8, 12 \\
 5, 7, 8, 12 &\rightarrow 5+7, 8, 12 \\
 8, 12, 12 &\rightarrow 8+12, 12 \\
 &12, 20
 \end{aligned}$$

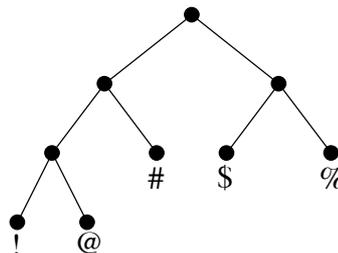
Potom algoritmus konštruuje spätne binárny strom, začínajúc s dvojčlennou postupnosťou 12, 20, ako to ukazuje obr. 72. Napríklad druhý strom vznikol z prvého nahradením vrcholu označeného 20 stromom na obr. 73, pretože 20 vzniklo súčtom 8 a 12. Výsledný optimálny Huffmanov kódovací strom získame nahradením každej frekvencie vo výstupnom strome znakom, ktorý má túto frekvenciu (obr. 74). Všimnime si, že toto nie je jediný možný optimálny kódovací strom. ■



Obr. 72



Obr. 73



Obr. 74

Veta. *Nech binárny strom, ktorého výška je h , má t koncových vrcholov. Potom*

$$\log_2 t \leq h.$$

Dôkaz. Matematickou indukciou vzhľadom na h dokážeme ekvivalentnú nerovnosť

$$t \leq 2^h.$$

Ak $h = 0$, binárny strom pozostáva z jediného vrcholu. V tomto prípade $t = 1$ a teda dokazovaná nerovnosť platí.

Predpokladajme, že tvrdenie vety platí pre všetky binárne stromy, ktorých výška je menšia ako h . Nech T je binárny strom, ktorý má t koncových vrcholov a výšku $h > 0$.

Všimnime si najprv prípad, keď koreň stromu T má iba jedného syna. Ak odstránime z tohto stromu koreň, získame nový binárny strom, ktorého výška je $h - 1$ a ktorý má ten istý počet koncových vrcholov ako strom T . Preto podľa indukčného predpokladu $t \leq 2^{h-1}$. Keďže $2^{h-1} < 2^h$, tvrdenie vety v tomto prípade platí.

Uvažujme teraz o druhom prípade, keď koreň stromu T má dvoch synov v_1 a v_2 . Nech T_i ,

$i \in \{1, 2\}$, je podstrom stromu T s koreňom v_i a má výšku h_i a t_i koncových vrcholov. Podľa indukčného predpokladu

$$t_i \leq 2^{h_i}, \quad i \in \{1, 2\}.$$

Koncové vrcholy stromu T sú iba koncové vrcholy stromov T_1 a T_2 , preto

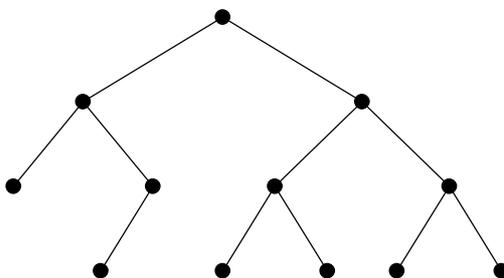
$$t = t_1 + t_2.$$

Z týchto dvoch vzťahov dostaneme

$$t = t_1 + t_2 \leq 2^{h_1} + 2^{h_2} \leq 2^{h-1} + 2^{h-1} = 2^h.$$



Príklad. Binárny strom na obr. 75 má výšku $h = 3$ a počet koncových vrcholov je $t = 6$. Zrejme platí $\log_2 t = \log_2 6 \leq 3 = h$. ■

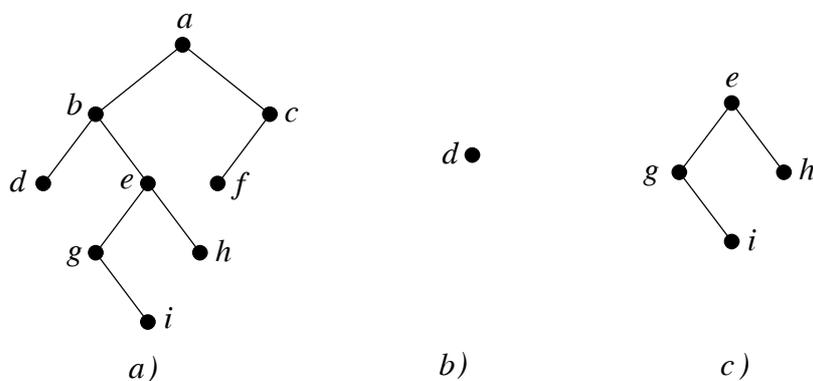


Obr. 75

Binárne stromy sú vhodné na reprezentáciu usporiadania na množinách dát. Štruktúra, ktorá sa tu používa, sa nazýva binárny prehľadávací strom.

Nech u je vnútorný vrchol binárneho stromu T . **Pravý** resp. **ľavý podstrom vrcholu** u je podstrom stromu T s koreňom v pravom resp. ľavom synovi vrcholu u .

Príklad. V binárnom strome na obr. 76a ľavým resp. pravým podstromom vrcholu b sú binárne stromy na obr. 76b resp. 76c. ■

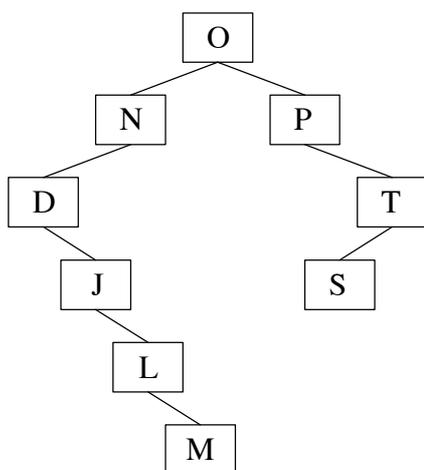


Obr. 76

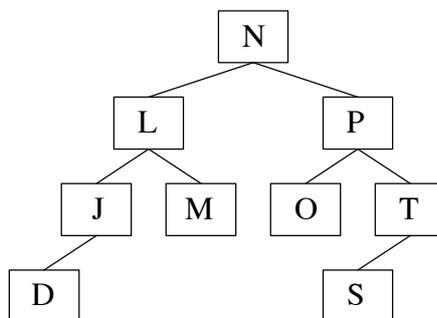
Nech A je konečná usporiadaná množina. **Binárny prehľadávací strom** je binárny strom T , v ktorom každému vrcholu je priradený práve jeden prvok množiny A . Toto priradenie je také, že pre každý vrchol u v T sú prvky priradené vrcholom ľavého podstromu vrcholu u menšie ako prvok priradený vrcholu u a tento prvok je menší ako prvky priradené vrcholom pravého podstromu vrcholu u .

Príklad. Prvky množiny $A = \{O, P, N, D, T, J, L, S, M\}$ s abecedným usporiadaním môžeme umiestniť do binárneho prehľadávacieho stromu (priradiť jeho vrcholom) napríklad tak, ako je to na obr. 77. Samozrejme, nie je to jediný spôsob. Iné umiestnenie prvkov množiny A do binárneho prehľadávacieho stromu vidieť na obr. 78. ■

Binárny prehľadávací strom T na obr. 77 bol skonštruovaný nasledujúcim spôsobom. Prezeráme prvky množiny A v poradí, v akom sú uvedené v zozname, t.j. najprv O , potom P , N atď. Súčasne vytvárame vrcholy stromu T , ktorým priraďujeme prvky množiny A v uvedenom poradí. Na začiatku vytvoríme vrchol, do ktorého umiestnime prvý prvok O , a tento vrchol označíme ako koreň. Pri tvorbe ďalších vrcholov a tiež hrán stromu T postupujeme nasledovne. Vezmeme v poradí ďalší prvok množiny A , do stromu T pridáme hranu a vrchol, do ktorého umiestnime tento prvok. Pri rozhodovaní, kam pridať hranu a vrchol začíname od koreňa. Ak zaradovaný prvok je menší (pri abecednom usporiadaní) ako prvok uložený v koreni, presunieme sa do ľavého syna, a ak zaradovaný prvok je väčší ako prvok v koreni, presunieme sa do pravého syna koreňa. Ak tento syn neexistuje, vytvoríme ho, pridáme hranu spájajúcu nový vrchol s koreňom a vložíme do nového vrcholu zaradovaný prvok množiny A . Ak tento syn existuje (označme ho v), opakujeme predchádzajúci postup s jedinou zmenou – úlohu koreňa tu prevezme vrchol v . Po umiestnení prvku do stromu T vezmeme ďalší prvok množiny A , porovnáme ho s koreňom (presnejšie s prvkom uloženým v koreni), podľa výsledku porovnania sa presunieme vľavo alebo vpravo, porovnáme ho s novým vrcholom, presunieme sa vľavo alebo vpravo a tak ďalej, až nakoniec vytvoríme novú hranu a vrchol, do ktorého vložíme vybraný prvok. Takýmto spôsobom uložíme všetky prvky množiny A do stromu a tak vytvoríme binárny prehľadávací strom. Celý postup je presnejšie opísaný v nasledujúcom algoritme.



Obr. 77



Obr. 78

Konstruktia binárneho prehľadávacieho stromu

Tento algoritmus skonštruuje binárny prehľadávací strom konečnej usporiadanej množiny A , $A \neq \emptyset$.

Vstup: Prostá postupnosť všetkých prvkov množiny A .

Výstup: Binárny prehľadávací strom T .

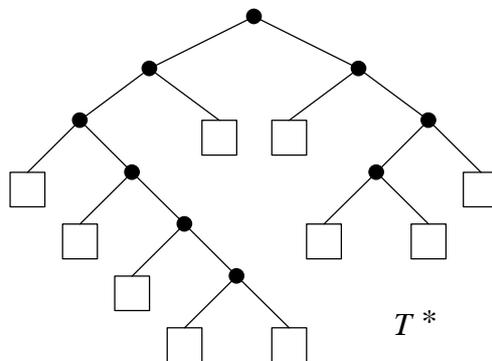
1. [Vytvor koreň] Nech w je prvý prvok danej postupnosti a nech T je strom s jedným vrcholom, ktorý je koreňom stromu T , a žiadnou hranou. Do koreňa vlož prvok w .
2. [Vezmi ďalší prvok] Nech w je ďalší člen postupnosti. Ak taký neexistuje, tak skonči.
3. [Začiatok prehľadávania] Nech v je koreň stromu T .
4. [Koniec prehľadávania] Ak prvok w je menší ako prvok vo vrchole v a v nemá ľavého syna, pridaj vrcholu v ľavého syna, vlož doň prvok w a choď na bod 2. Ak w je väčší ako prvok vo v a v nemá pravého syna, pridaj vrcholu v pravého syna, vlož doň prvok w a choď na bod 2.
5. [Pokračovanie prehľadávania] Ak prvok w je menší ako prvok vo v , vykonaj
 $v :=$ ľavý syn vrcholu v
a choď na bod 4. Ak w je väčší ako prvok vo v , vykonaj
 $v :=$ pravý syn vrcholu v
a choď na bod 4.

Binárne prehľadávacie stromy sú užitočné pri lokalizovaní dát, t.j. zistení, či dátová položka D sa nachádza v danom dátovom súbore, a ak áno, na ktorom mieste. V prípade, že dátový súbor je uložený v binárnom prehľadávacom strome, začíname pri lokalizácii v koreni prehľadávacieho stromu a postupujeme opakovaným porovnávaním položky D s položkou v aktuálnom vrchole prehľadávacieho stromu. Ak D sa rovná dátovej položke v aktuálnom vrchole, znamená to, že sme našli položku D v prehľadávacom strome. Ak D je menšie ako položka v aktuálnom vrchole v , stane sa aktuálnym vrcholom ľavý syn (pokiaľ existuje) vrcholu v a opakujeme predchádzajúci postup. Ak D je väčšia ako položka v aktuálnom vrchole, aktuálnym vrcholom sa stane pravý syn (pokiaľ existuje) vrcholu v a opakujeme predchádzajúci postup. Ak tento syn vrcholu v neexistuje, znamená to, že dátová položka sa v danom dátovom súbore nenachádza.

Čas potrebný na vyhľadanie dátovej položky v binárnom prehľadávacom strome je najdlhší vtedy, keď daná položka sa nenachádza v prehľadávacom strome a pri jej lokalizácii sme prešli od koreňa najdlhšiu cestu v strome. Takže najdlhší čas potrebný na lokalizáciu dátovej položky v binárnom prehľadávacom strome je úmerný výške tohto stromu. Prehľadávanie stromu je teda tým kratšie, čím menšia je jeho výška. Je známych viac spôsobov ako minimalizovať výšku binárneho prehľadávacieho stromu [10].

Urobíme teraz odhad času potrebného na lokalizáciu dát v binárnom prehľadávacom strome v najhoršom prípade (čas prehľadávania je najdlhší) v závislosti od počtu jeho vrcholov, t.j. od počtu prvkov dátového súboru uložených v tomto strome. Tento čas budeme merať počtom vykonaných porovnaní hľadanej dátovej položky s položkami uloženými v prehľadávacom strome. Nech T je binárny prehľadávací strom s n vrcholmi a nech T^* je úplný binárny strom, ktorý vznikne z T pridaním ľavého a pravého syna k existujúcim vrcholom všade tam, kde taký syn neexistuje. Na obr. 79 je takto vytvorený úplný binárny strom T^* k binárnemu prehľadávaciemu stromu T na obr. 77. Pridané vrcholy sú tu nakreslené ako štvorčeky. Čas v najhoršom prípade potrebný na lokalizáciu dátovej položky sa rovná počtu porovnaní tejto položky s položkami uloženými vo vrcholoch najdlhšej cesty stromu T so

začiatočným vrcholom v koreni. To je však práve výška stromu T^* . Medzi výškou h stromu T^* a počtom t jeho koncových vrcholov platí vzťah $\log_2 t \leq h$. Úplný binárny strom T^* má n vnútorných vrcholov, počet jeho koncových vrcholov je teda $t = n + 1$. Odtiaľ dostaneme $h \geq \log_2(n + 1)$. Dokázali sme nasledujúcu vetu.



Obr. 79

Veta. Pre čas h (počet vykonaných porovnaní) v najhoršom prípade potrebný na lokalizáciu dátovej položky v binárnom prehľadávacom strome s n vrcholmi platí

$$h \geq \log_2(n + 1).$$



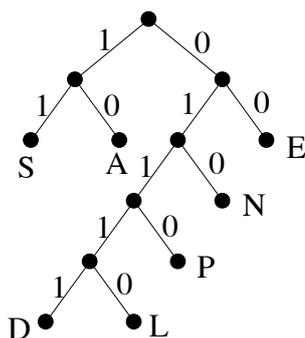
Dá sa dokázať, že ak výška binárneho prehľadávacieho stromu je minimalizovaná, tak platí $h = \lfloor \log_2(n + 1) \rfloor$, kde $\lfloor x \rfloor$ znamená celú časť čísla x . Tak napríklad, keďže

$$\log_2(2\,000\,000 + 1) = 20,93156929067\dots$$

je možné uložiť 2 milióny položiek do binárneho prehľadávacieho stromu tak, že na vyhľadanie položky alebo určenie, že sa v tomto strome nenachádza, je potrebných najviac 21 porovnaní.

Cvičenia

- Nakreslite úplný binárny strom, ktorý má požadované vlastnosti alebo vysvetlite, prečo taký graf neexistuje.
 - 4 vútorné vrcholy, 5 koncových vrcholov,
 - 9 koncových vrcholov, výška je 3,
 - 9 koncových vrcholov, výška je 4.
- Dekódujte bitový reťazec
 - 011000010
 - 01110100110
 - 01111001001110
 - 1110011101001111
 použitím Huffmanovho kodu určeného binárnym stromom na obr. 80.
- Zakódujte slová SANE, PENALE, NASADENA, NEDELNA pomocou Huffmanovho kódu z obr. 80.



Obr. 80

4. Zostrojme optimálny Huffmanov kód pre znaky s požadovanou frekvenciou podľa tabuľky

znak	frekvencia
α	5,00
β	6,00
γ	6,00
δ	11,00
ε	20,00

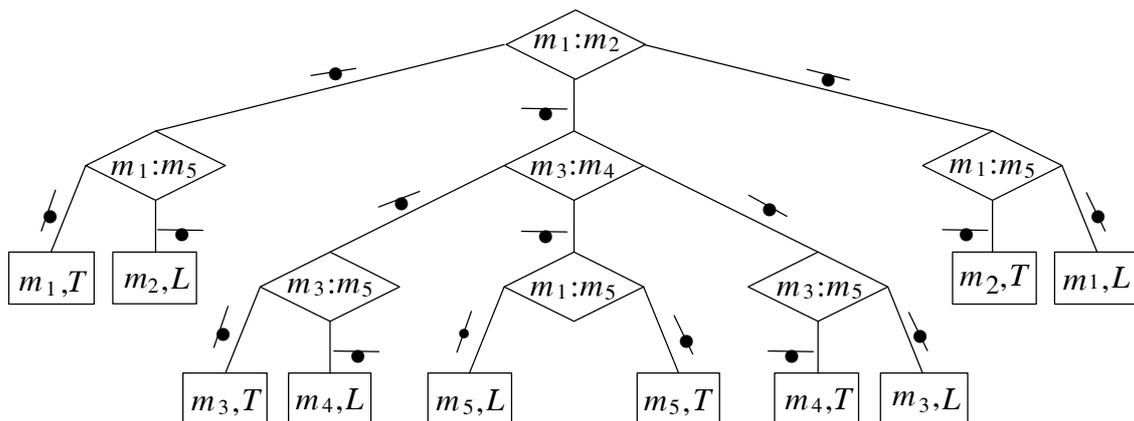
5. Aká je maximálna výška úplného binárneho stromu, ktorý má t koncových vrcholov?
6. Priradte slová KAŽDÝ ČLOVEK VIAC ALEBO MENEJ POCÍŤUJE POTREBU STAŤ SA INÝM ČLOVEKOM vrcholom binárneho prehľadávacieho stromu pri abecednom usporiadaní slov.
7. Rozhodnite, či je pravdivé tvrdenie: Nech T je binárny strom a nech pre všetky vrcholy v stromu T platí, že dátová položka vo v je väčšia ako dátová položka v jeho ľavom synovi ale je menšia ako dátová položka v jeho pravom synovi. Potom T je binárny prehľadávací strom.

Rozhodovacie stromy

Postup riešenia mnohých úloh je možné rozdeliť do niekoľkých fáz, z ktorých každá má rovnaký charakter: najprv sa vykoná určitá činnosť, súvisiaca s danou úlohou, ktorá môže mať niekoľko čiastkových výsledkov. Podľa toho, ktorý z výsledkov nastal, dostáva sa riešenie úlohy do ďalšej fázy. Takéto spôsoby riešenia úloh je možné znázorniť pomocou koreňových stromov. Každéj fáze riešenia zodpovedá jeden vnútorný vrchol, v ňom je zaznamenaná činnosť, ktorá sa v tejto fáze vykonáva, a toľko z neho vychádzajúcich hrán, koľko je v tejto fáze možných čiastkových výsledkov. Každá z týchto hrán zodpovedá práve jednému z možných

výsledkov a inciduje s ďalším vrcholom, ktorý opisuje činnosť v nasledujúcej fáze riešenia. Koncové vrcholy stromu zodpovedajú možným výsledkom úlohy. Koreňové stromy, ktoré uvedeným spôsobom opisujú riešenia úloh sa nazývajú *rozhodovacie stromy*.

Príklad. (Problém jednej falošnej mince z piatich) Z piatich mincí, ktoré na pohľad vyzerajú rovnako, má práve jedna inú hmotnosť ako ostatné mince. Vážením na rovnoramenných váhach bez použitia závaží je treba rozhodnúť, ktorá z mincí je falošná (t.j. inej hmotnosti ako ostatné) a či je ľahšia alebo ťažšia ako pravá minca.



Obr. 81

Riešenie. Algoritmus riešenia tejto úlohy je opísaný pomocou rozhodovacieho stromu na obr. 81. Mince označíme m_1, m_2, m_3, m_4, m_5 . Začneme vykonávať činnosť zapísanú v koreni - $m_1 : m_2$, čo znamená, že máme položiť mincu m_1 na ľavú a mincu m_2 na pravú miskú váh. Hrana označená znakom



znamená, že minca na ľavej strane je ťažšia ako minca na pravej strane váh. Podobne hrana označená znakom



znamená, že minca na ľavej strane je ľahšia ako minca na pravej strane a hrana označená znakom



vyjadruje, že mince na oboch stranách majú rovnakú hmotnosť. Napríklad, ak porovnáme mince m_1 a m_2 (ako je to zaznačené v koreni) a ľavá strana bude ťažšia, znamená to, že buď minca m_1 je ťažšia ako ostatné alebo minca m_2 je ľahšia ako ostatné. V tomto prípade, ako to vidieť z rozhodovacieho stromu, porovnáme hmotnosti mincí m_1 a m_5 (táto minca je pravá) a okamžite rozhodneme, či falošná minca je m_1 alebo m_2 a či je ľahšia alebo ťažšia. Výsledok

je uvedený v koncovom vrchole. Napríklad, ak pri porovnávaní hmotností mincí m_1 a m_5 sa jazýček váh vychýli, znamená to, že falošná minca je m_1 (má inú hmotnosť ako pravá minca m_5) a je ťažšia. V rozhodovacom strome je tento výsledok zobrazený tak, že hrana s označením \bullet inciduje s koncovým vrcholom, v ktorom je uvedený výsledok m_1, T . ■

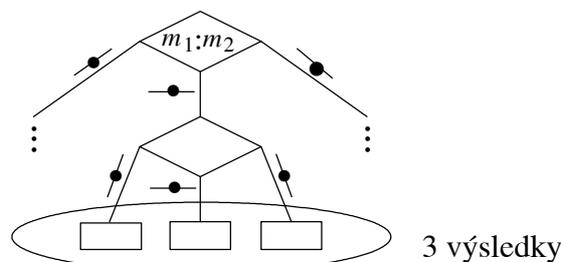
Definujme čas potrebný na riešenie problému falošnej mince ako počet vykonaných vážení. Potom čas riešenia tohto problému v najhoršom prípade (vykoná sa najväčší počet vážení) sa rovná výške rozhodovacieho stromu, ktorý opisuje algoritmus riešenia tohto problému. V prípade algoritmu opísaného rozhodovacím stromom na obr. 81 sa tento čas rovná 3.

Rozhodovacie stromy môžeme použiť aj na dôkaz toho, že algoritmus z obr. 81 je optimálny, t.j. že žiadny iný algoritmus, ktorý rieši problém falošnej mince z piatich, nemá čas v najhoršom prípade menší ako 3. Za tým účelom predpokladajme, že existuje algoritmus, ktorý rieši tento problém v čase v najhoršom prípade menšom alebo rovnom 2. Tento algoritmus sa dá opísať pomocou rozhodovacieho stromu a keďže čas v najhoršom prípade sa rovná najviac 2, je výška tohto stromu najviac 2. Pretože každý vnútorný vrchol tohto stromu má najviac troch synov, je počet jeho koncových vrcholov najviac $3^2 = 9$. Každý koncový vrchol zodpovedá jednému z možných výsledkov problému, takže rozhodovací strom, ktorého výška je najviac 2, opíše najviac 9 možných výsledkov. Avšak problém falošnej mince z piatich má 10 možných výsledkov:

$$\begin{aligned} & m_1, L; m_2, L; m_3, L; m_4, L; m_5, L; \\ & m_1, T; m_2, T; m_3, T; m_4, T; m_5, T. \end{aligned}$$

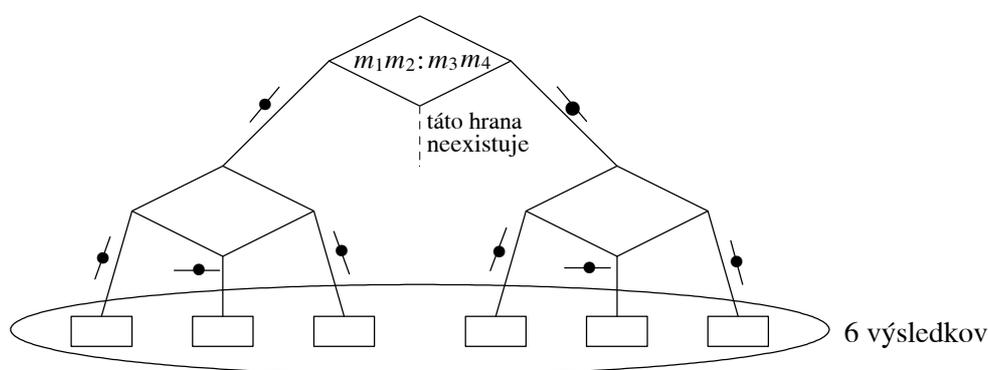
To je spor. Preto žiadny algoritmus riešiaci problém falošnej mince z piatich nemá čas v najhoršom prípade menší ako 3.

Ukázali sme si, ako sa dajú rozhodovacie stromy použiť na určenie dolnej hranice času v najhoršom prípade potrebného na riešenie problému. Stáva sa však, že takto určenú dolnú hranicu pre daný problém žiadny algoritmus, ktorý daný problém rieši, nemôže dosiahnuť. Tak je to napríklad v prípade falošnej mince zo štyroch. Keďže problém falošnej mince zo štyroch má 8 možných výsledkov, mohol by sa algoritmus riešiaci tento problém dať zapísať do rozhodovacieho stromu výšky 2. Avšak podrobnejšie skúmanie ukáže, že je potrebný rozhodovací strom výšky aspoň 3. Pri prvom vážení máme 2 možnosti. Buď na misky váh položíme po jednej minci alebo po dvoch minciach.



Obr. 82

Uvažujme prvý prípad. Z obr. 82 je vidieť, že keď porovnáme hmotnosť mincí m_1 a m_2 a váhy zostanú v rovnováhe, poskytuje rozhodovací strom už len tri výsledky. Avšak v tejto situácii, keď sme našli 2 pravé mince, sú možné až 4 výsledky: m_3, L ; m_3, T ; m_4, L ; m_4, T . To znamená, že žiadny algoritmus začínajúci porovnaním dvoch mincí nemôže vyriešiť tento problém na 2 alebo menej vážení v najhoršom prípade. Obrázok 54 zase ilustruje druhý prípad, keď porovnáваме dvojice mincí. V tomto prípade váhy nemôžu zostať v rovnováhe (falošná minca sa nachádza na niektorej miske). To však znamená, že rozhodovací strom umožní len 6 výsledkov, avšak problém má v tejto situácii stále 8 možných výsledkov. Vidíme, že ani algoritmy, ktoré začínajú porovnávať hmotnosť dvojice mincí, nevyriešia tento problém na 2 alebo menej vážení v najhoršom prípade. Takže každý algoritmus, ktorý rieši problém falošnej mince zo štyroch, vyžaduje aspoň 3 vážení v najhoršom prípade.



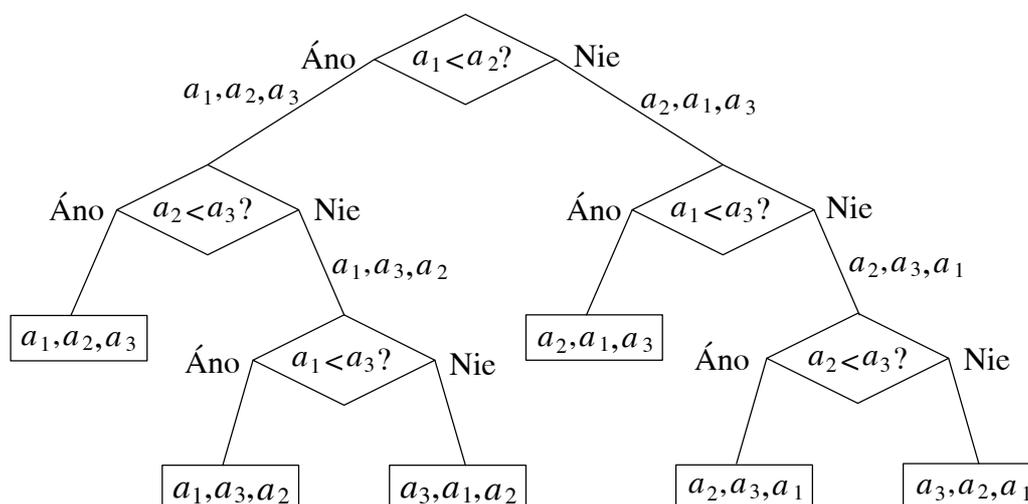
Obr. 83

Venujme sa teraz ďalšiemu problému - triedeniu, ktoré môžeme charakterizovať takto: je daný súbor n prvkov x_1, x_2, \dots, x_n lineárne usporiadanej množiny. Je potrebné zoradiť tieto prvky v poradí od najmenšieho po najväčší alebo obrátene. Obmedzíme sa na triediaci algoritmus, ktorý opakovane porovnáva dva prvky a v závislosti od výsledku porovnania modifikuje pôvodný súbor.

Príklad. Jeden triediaci algoritmus, ktorý usporiada tri navzájom rôzne prvky a_1, a_2, a_3 podľa veľkosti, je daný rozhodovacím stromom na obr. 84. Každý vnútorný vrchol stromu predstavuje porovnanie niektorých dvoch prvkov a_i, a_j (vo vrchole je to znázornené otázkou $a_i < a_j?$), podľa výsledku porovnania prípadnú vzájomnú výmenu prvkov a_i, a_j v súbore tak, aby menší prvok bol vľavo od väčšieho. Pre väčšiu názornosť je takto upravený súbor pripísaný k hrane (označená je tiež symbolom Áno alebo Nie, podľa toho, či situácia $a_i < a_j$ nastala alebo nie), ktorá vchádza do vrcholu určujúceho ďalšiu činnosť. V koncovom vrchole je už súbor, v ktorom sú prvky zoradené podľa veľkosti.

Definujme čas triedenia ako počet vykonaných porovnaní. Potom čas triedenia v najhoršom prípade sa rovná výške rozhodovacieho stromu. Pre triediaci algoritmus daný rozhodovacím stromom na obr. 84 sa tento čas rovná 3. Ukážeme, že tento algoritmus je optimálny, t.j. že neexistuje algoritmus, ktorý by usporiadal tri prvky v najhoršom prípade za čas menší ako 3. Dokážeme to nepriamo. Predpokladajme, že taký algoritmus existuje. Tento algoritmus sa potom dá opísať rozhodovacím stromom, ktorého výška je najviac 2. Keďže v tomto strome každý vnútorný vrchol má najviac 2 synov, má tento strom najviac 4 koncové

vrcholy. Tieto koncové vrcholy zodpovedajú 4 výsledkom. Avšak náš problém má až 6 možných výsledkov. To je spor. Preto algoritmus opísaný rozhodovacím stromom na obr. 84 je optimálny. ■



Obr. 84

Nasledujúca veta udáva dolné ohraničenie času triedenia v najhoršom prípade pre n prvkový súbor.

Veta. Pre počet $p(n)$ porovnaní potrebných v najhoršom prípade na usporiadanie n prvkov, $n \geq 1$, platí

$$\frac{n \log_2 n}{4} \leq p(n).$$

Dôkaz. Nech T je rozhodovací strom, ktorý popisuje triediaci algoritmus pre n prvkov. Ak označíme jeho výšku h , tak

$$h = p(n).$$

Triediaci algoritmus pre n prvkov má $n!$ možných výsledkov, čo sa rovná počtu koncových vrcholov stromu T . T je binárny strom, preto medzi jeho výškou a počtom jeho koncových vrcholov platí vzťah

$$\log_2 n! \leq h = p(n).$$

Nech $\lfloor x \rfloor$ označuje celú časť čísla x . Potom platí

$$\begin{aligned} n! &= n(n-1)(n-2)\dots 2 \cdot 1 \geq \\ &\geq n(n-1)(n-2)\dots \left(\left\lfloor \frac{n}{2} \right\rfloor + 1\right) \geq \\ &\frac{n}{2} \frac{n}{2} \frac{n}{2} \dots \frac{n}{2} = \left(\frac{n}{2}\right)^{\lfloor \frac{n}{2} \rfloor + 1} \geq \left(\frac{n}{2}\right)^{\frac{n}{2}} \end{aligned}$$

a

$$\begin{aligned} p(n) &\geq \log_2 n! \geq \log_2 \left(\frac{n}{2}\right)^{\frac{n}{2}} = \frac{n}{2} \log_2 \left(\frac{n}{2}\right) = \\ &= \frac{n}{2} (\log_2 n - \log_2 2) = \frac{n}{2} (\log_2 n - 1). \end{aligned}$$

Teraz ukážeme, že pre $n \geq 4$

$$\frac{n \log_2 n}{4} \leq \frac{n}{2} (\log_2 n - 1),$$

čo je ekvivaletné s

$$\frac{1}{2} \leq \frac{\log_2 n}{4}.$$

Táto nerovnosť, a tým aj predchádzajúca s ňou ekvivaletná, je pre $n \geq 4$ pravdivá, lebo $\log_2 4 = 2$ a $\log_2 x$ je rastúca funkcia.

Tým sme dokázali platnosť nerovnosti

$$\frac{n \log_2 n}{4} \leq p(n)$$

pre $n \geq 4$. Keďže

$$p(1) = \log_2 1! = 0 = \frac{1 \cdot \log_2 1}{4},$$

$$p(2) = \log_2 2! = 1 > \frac{1}{2} = \frac{2 \log_2 2}{4},$$

$$p(3) = \log_2 3! = \log_2 6 = 1 + \log_2 3 > \frac{3 \log_2 3}{4},$$

platí tá nerovnosť pre všetky $n \in \mathbb{N}^+$.



Cvičenia

1. Nakreslite rozhodovací strom opisujúci algoritmus, ktorý rieši problém jednej falošnej mince z dvanástich najviac na tri váženia.
2. Zo štyroch mincí, ktoré sú na pohľad rovnaké, práve jedna je falošná - má inú hmotnosť ako ostatné. Nakreslite rozhodovací strom opisujúci algoritmus, ktorý určí falošnú mincu v najhoršom prípade najviac na 2 váženia na rovnoramenných váhach bez použitia závaží. Nie je potrebné určiť, či falošná minca je ľahšia alebo ťažšia ako pravé mince.
3. Ukážte, že na riešenie problému z cvičenia 2 sú potrebné aspoň dve váženia.
4. Pomocou rozhodovacieho stromu určte dolné ohraničenie počtu porovnaní v najhoršom prípade potrebných na usporiadanie 5 prvkov podľa veľkosti. Nájdite triediaci algoritmus pre 5 prvkov, ktorý v najhoršom prípade potrebuje práve tento počet porovnaní.