Přehled přednášky

Kapitola 6. Pojem grafu

- stupeň vrcholu v grafu
- podgrafy a izomorfismus
- orientované grafy a multigrafy
- implementace grafů

Kapitola 7. Souvislost grafu

- motivace
- spojení mezi vrcholy, komponenty grafu
- prohledávání grafu
- vyšší stupně souvislosti
- Eulerovské grafy „jedním tahem“

Opakování

- graf jako algebraická struktura

  Definice 6.1. *Graf G* (*jednoduchý* graf, *obyčejný* graf) je uspořádaná dvojice $G = (V, E)$, kde V je množina *vrcholů* a E je množina *hran* – systém dvouprvkových podmnožin množiny V.

- třídy grafů K_n, $K_{m,n}$, C_n, P_n
- stupeň vrcholu $\deg_G(x)$

$$\deg_G(x) = |\{e \in E : x \in e\}|$$

- Věta 6.3. (alias V 1.1.)
  Součet stupňů v grafu je roven dvojnásobku počtu hran:

$$\sum_{v \in V(G)} deg_G(v) = 2|E(G)|.$$

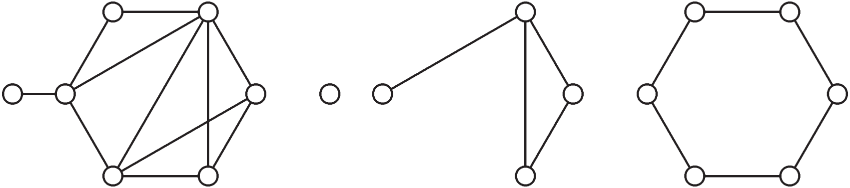
6.3. Podgrafy a izomorfismus

Často uvažujeme případy, kdy z daného grafu vynecháme některé vrcholy (a hrany s nimi incidentní), nebo vynecháme hrany, případně obojí. Dostaneme tzv. *podgraf*.

Definice Graf H nazveme *podgrafem* grafu G jestliže $V(H) \subseteq V(G)$ a $E(H) \subseteq E(G)$. Píšeme $H \subseteq G$.

Všimněte si: vynecháme-li z G některý vrchol, musíme současně vynechat i všechny hrany s ním incidentní. Definice je v tomto smyslu korektní, neboť pokud bychom vynechali jen vrcholy a nikoli hrany, H by nebyl graf.

Příklad:

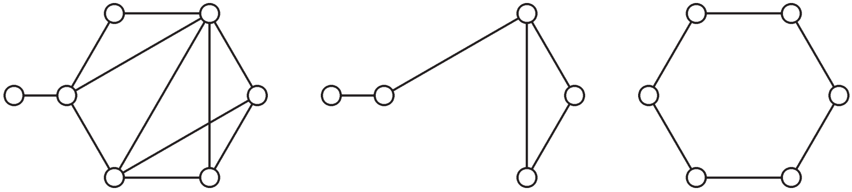


Obrázek: *Graf G a jeho podgrafy H_1 a H_2.*

Indukovaný podgraf

Podgraf I grafu G nazveme *indukovaným* podgrafem grafu G jestliže $E(I)$ obsahuje všechny hrany grafu G, které jsou incidentní s vrcholy z $V(I)$ (vynecháme pouze hrany, které byli incidentní s vynechanými vrcholy).

Příklad:



Obrázek: *Graf G a podgrafy H_1 (indukovaný) a H_2 (není indukovaný).*

Faktor grafu

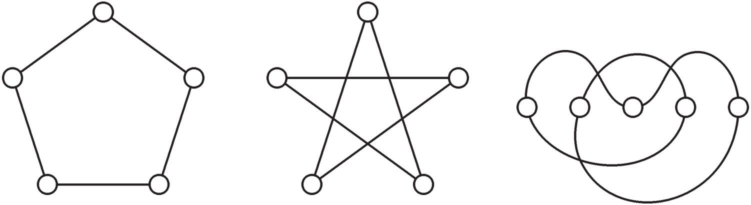
Podgraf F grafu G nazveme *faktorem* grafu G jestliže $V(F) = V(G)$.

Definice 6.6. *Isomorfismus grafů G a H*
je bijektivní zobrazení $f : V(G) \to V(H)$, pro které platí, že každé dva vrcholy $u, v \in V(G)$ jsou spojené hranou v G právě tehdy, když jsou $f(u), f(v)$ spojené hranou v H.

Stručně

$$\forall u, v \in V(G) : uv \in E(G) \Leftrightarrow f(u)f(v) \in E(H).$$

Příklad:



Obrázek: *Isomorfní grafy.*

Zřejmé: Isomorfní grafy G a H mají

- stejný počet vrcholů
- stejný počet hran
- každý podgraf grafu G musí být podgrafem grafu H a naopak
- stejný nejvyšší stupeň $\Delta(G) = \Delta(H)$
- stejný nejnižší stupeň $\delta(G) = \delta(H)$
- stejnou stupňovou posloupnost
- ...

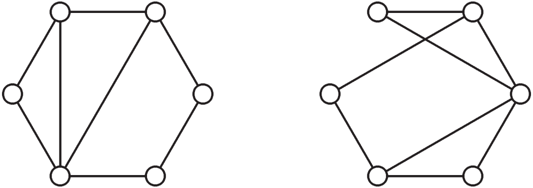
Pokud je bijekce f isomorfismem, musí zobrazit na sebe vrcholy stejných stupňů, tj.

$$\deg_G(v) = \deg_H(f(v)).$$

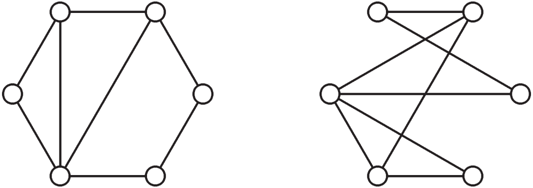
Opačná implikace neplatí! Nestačí porovnat stupňové posloupnosti!

Příklad:

Jsou následující dva grafy se stupňovou posloupností $(2, 2, 2, 3, 3, 4)$ isomorfní?



Obrázek: *Jsou tyto dva grafy isomorfní?*



Obrázek: *Jsou tyto dva grafy isomorfní?*

Poznámka

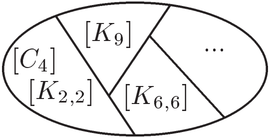
Uvedené příklady nám ukazují některé postupy, jak poznat zda grafy jsou nebo nejsou isomorfní.
Žádný „rozumný“ univerzální postup pro nalezení isomorfismu není znám!
Jediný (známý) univerzální postup pro nalezení či vyvrácení isomorfismu mezi dvěma grafy je typu *vyzkoušejte všechny* bijekce mezi vrcholy těchto grafů, kterých je $n!$.

Věta 6.8. Relace „být isomorfní“ $\simeq$ je relací ekvivalence na třídě všech grafů.

Důkaz
Relace $\simeq$ je

- reflexivní, protože graf je isomorfní sám sobě identickým zobrazením,
- symetrická, neboť k isomorfismu (bijekci) f existuje (jednoznačně) f^{-1},
- tranzitivní, neboť skládáním zobrazení (isomorfismů) dostaneme opět zobrazení (isomorfismus).

Pokud mluvíme o nějakém *grafu*, myslíme tím obvykle jeho celou třídu isomorfismu, nezáleží na konkrétní reprezentaci, nakreslení či pojmenování grafu.



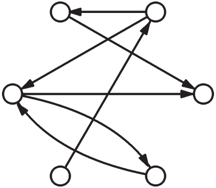
Obrázek: *Třídy grafů.*

6.4. Orientované grafy a multigrafy

Pokud graf reprezentuje dopravní síť, silniční síť nebo jiný produktovod, může být užitečné zavést *orientaci hrany*. Rozlišujeme *výchozí* a *koncový* vrchol. Hrana uv není totožná s hranou vu.

Orientovaným grafem rozumíme uspořádanou dvojici $G = (V, E)$, kde V je množina *vrcholů* a množina *hran* je $E \subseteq V \times V$.

V nakreslení znázorníme orientované hrany šipkami.



Obrázek: *Orientovaný graf.*

Poznámka

Obecnější pojem než orientovaný graf, je *multigraf*. Povolíme vícenásobné hrany a smyčky.

6.5. Implementace grafů

Vrcholy grafy G označíme přirozenými čísly $0, 1, \ldots, n-1$. Mezi nejběžnější implementace grafů v počítači patří:

- *maticí sousednosti*: dvourozměrné pole `g[][]`
  kde `g[i][j]=1` pokud graf obsahuje hranu ij, jinak `g[i][j]=0`
  Výhody: jednoduché, snadno ověříme existenci hrany uv

- *výčtem sousedů*: dvourozměrné pole `h[][]`, pole `d[]` stupňů vrcholů
  kde `h[i][0]`, `h[i][1]`, `...`, `h[i][d[i]-1]` udávají seznam sousedů vrcholu i
  Výhody: vhodné pro grafy s malým počtem hran, snadno určíme $\deg(v)$

Je třeba dbát na symetrii hran ij a ji!

Obě reprezentace vhodné i pro orientované grafy.
Pro multigrafy se hodí jen zadání výčtem sousedů.

Složitější datové struktury pro ohodnocené grafy
(přiřadíme ohodnocení hran/vrcholů).
Můžeme použít další pole, strukturované datové typy...

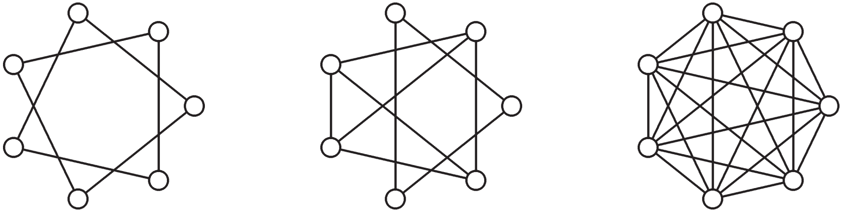
7. Souvislost grafu

Pokud graf reprezentuje počítačovou, nebo dopravní síť je přirozené zjistit zda a jak je možno cestovat (vyslat signál) z vrcholu u do vrcholu v. Zavedeme pojem *souvislost grafu*.

Podobně má smysl zkoumat odolnost sítě vůči lokálním výpadkům:

- redundance uzlů
- dosažitelnost cílového vrcholu i při přerušení některých hran

Dostáváme se k pojmu *stupeň vrcholové a hranové souvislosti*.



Obrázek: *Souvislé a nesouvislé grafy.*

7.1. Spojení vrcholů, komponenty

Neformálně: Graf je souvislý, pokud existuje „spojení“ mezi každými dvěma vrcholy (byť nepřímé).

Formálně zavedeme pojmy *sled*, *tah* a *cesta* v grafu.

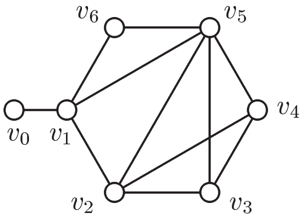
Definice

Sledem $v_0 v_n$ v grafu G rozumíme takovou posloupnost vrcholů a hran

$$v_0, e_1, v_1, e_2, v_2, \ldots, e_n, v_n,$$

kde v_i jsou vrcholy grafu G a e_i jsou hrany grafu G takové, že e_i má koncové vrcholy v_{i-1} a v_i. Počet hran n nazveme *délkou sledu* $v_0 v_n$.

Pokud nejsou násobné hrany, můžeme sled zadat jako posloupnost vrcholů.



sled $v_0, v_1, v_5, v_6, v_1, v_0, v_1, v_2$

sled $v_0, v_1, v_2, v_4, v_3, v_5, v_6$

sled v_4

Lemma 7.1. Zavedeme relaci $\sim$ na množině $V(G)$ grafu G tak, že $u \sim v$, právě když v grafu G existuje sled uv. Potom platí, že relace $\sim$ je relací ekvivalence.

Důkaz Reflexivita plyne ihned z existence triviálního sledu uu délky 0. Symetrie je zřejmá, neboť ke každému sledu uv v G sestavíme sled vu v G „obrácením" posloupnosti vrcholů a hran (neorientovaný graf). Tranzitivita plyne z faktu, že spojením sledu $u, \ldots, v$ a $v, \ldots, w$ dostaneme opět sled – sled $u, \ldots, w$. $\square$

Pojem souvislosti je vybudován na pojmu *sled*.
Někdy však není žádoucí při popisu dosažitelnosti, aby se opakovaly hrany či vrcholy (potrubí, dopravní sítě, elektrické sítě, ...).

Definice
Tah je sled ve kterém se neopakují žádné hrany (kreslíme „jedním tahem").
Cesta je sled ve kterém se neopakují ani žádné vrcholy.

Věta 7.2.

Pokud mezi dvěma vrcholy grafu G existuje sled, pak mezi nimi existuje cesta.

Důkaz Nechť S $u = v_0, e_1, v_1, \ldots, e_n, v_n = v$ je sled délky n mezi vrcholy u a v v G. Hledáme sled W z vrcholu u do vrcholu v, který bude cestou (žádný vrchol se nebude opakovat). Pokud se v S žádný vrchol neopakuje, je $W = S$ hledaným sledem.
Pokud se některé vrcholy opakují vezmeme první takový vrchol v_i. Vynecháme celý úsek sledu mezi prvním výskytem v_i a jeho *posledním* opakováním v_k. Získáme sled S', ve kterém se v_i již neopakuje.
Pokud se v S' žádný vrchol neopakuje, je $W = S'$. Jinak postup zopakujeme pro další vrchol.
Postup je jistě konečný, neboť v G je konečně mnoho vrcholů. □

Jestliže v G existuje sled uv, tak můžeme postupem z důkazu Věty 7.2. vyrobit ze sledu uv cestu uv a říkáme, že vrcholy u a v jsou v G *spojeny cestou*.

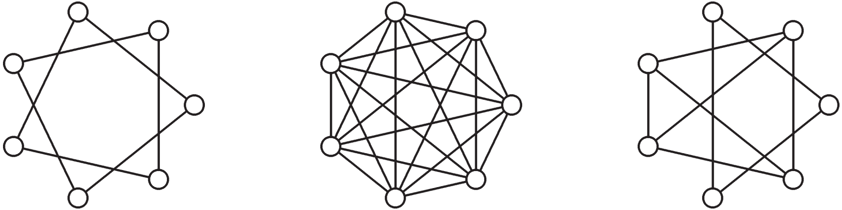
Nyní můžeme vyslovit definici souvislého grafu:

Definice

Relace $\sim$ z Lemmatu 7.1. definuje rozklad na množině $V(G)$. Třídy ekvivalence relace $\sim$ jsou podmnožiny $V(G)$ a podgrafy indukované na těchto podmnožinách se nazývají *komponenty souvislosti* grafu G.

Definice

Řekneme, že graf G je *souvislý*, pokud je G tvořený jedinou komponentou souvislosti, tj. pokud každé dva vrcholy G jsou spojeny cestou (sledem).



Obrázek: *Příklad souvislých grafů a nesouvislého grafu.*

7.2. Prohledávání grafu

Pro vytvoření obecného schématu algoritmu pro „procházení“ grafu vystačíme s několika málo datovými stavy a jednou pomocnou strukturou:

Vrchol má stavy ...

- iniciační – na začátku,
- nalezený – jakmile jej přes některou hranu najdeme,
- zpracovaný – jakmile prozkoumáme všechny hrany z něj vycházející.

Hrana má stavy ...

- iniciační – na začátku,
- zpracovaná – jakmile je prozkoumána z jednoho koncového vrcholu.

Úschovna je pomocná datová struktura (posloupnost/množina),

- ukládáme nalezené a zatím nezpracované vrcholy.

Způsob, jak vybíráme vrcholy z úschovny, určuje variantu algoritmu procházení grafu (do hloubky/do šířky). Pro vrcholy a hrany se pak provádí konkrétní programové akce – prohledání a zpracování grafu.

Algoritmus 7.4.

Procházení souvislých komponent grafu – všechny hrany

Na začátku

- vybereme libovolný vrchol
- vrcholům i hranám přiřadíme iniciační stav

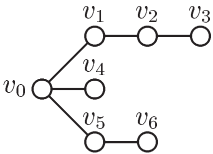
```
// na vstupu je graf G
vstup < graf G;
stav(všechny vrcholy a hrany G) = iniciační;
uschovna U = libovolný vrchol v0 grafu G;
stav(v0) = nalezený;
```

```
// zpracování vybrané komponenty G
while (U je neprázdná) {
   vyber vrchol v z úschovny U; U = U - v;
   ZPRACUJ(v);
   for (e hrany vycházející z v)      // pro všechny hrany
      if (stav(e) == iniciační)
      ZPRACUJ(e);
      w = druhý vrchol hrany e = vw;  // známe sousedy?
      if (stav(w) == iniciační) {
         stav(w) = nalezený;
         U = U sjednocení w;
      }
      stav(e) = zpracovaná;
   }
   stav(v) = zpracovaný;
   // případný přechod na další komponentu G
   if (U je prázdná && G má další komponenty)
      uschovna U = {vrchol v_1 z další komponenty G};
}
```

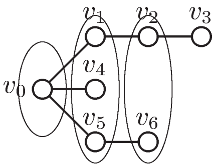
Různými implementacemi *procházení grafu* dostaneme různé algoritmy

- *Procházení „do hloubky"* – úschovnu `U` implementujeme jako zásobník, tj. dále prohledáváme od posledních nalezených vrcholů.
- *Procházení „do šířky"* – úschovna `U` implementujeme jako frontu, tj. dále prohledáváme od prvních nalezených vrcholů.
- *Dijkstrův algoritmus* pro nejkratší cestu – z úschovny vybíráme vždy vrchol nejbližší k počátečnímu `v_0`;
(jako prohledávání do šířky, kdy hrany mají „různé délky").

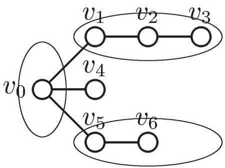
Příklad:



Obrázek: *Prozkoumejte graf postupem do hloubky i do šířky (od vrcholu v_0).*



Obrázek: *Prohledávání do šířky (od vrcholu v_0).*



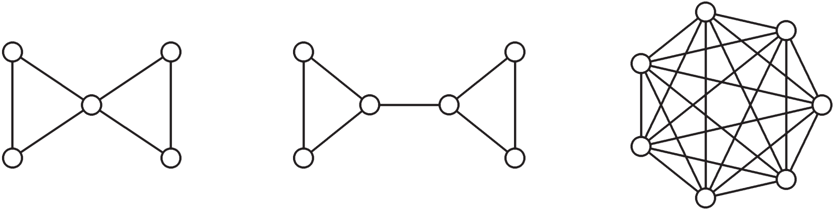
Obrázek: *Prohledávání do hloubky (od vrcholu v_0).*

7.3. Vyšší stupně souvislosti

V praxi nás může zajímat nejen, jestli existuje spojeni (cesta) mezi vrcholy grafu), ale také jestli bude existovat spojení v případě lokálních výpadků – odolnost a redundance (datová síť, železniční a silniční síť).

Definice Graf G je *hranově k-souvislý*, $k > 1$, pokud i po odebrání libovolných $k-1$ hran z G zůstane výsledný faktor souvislý.

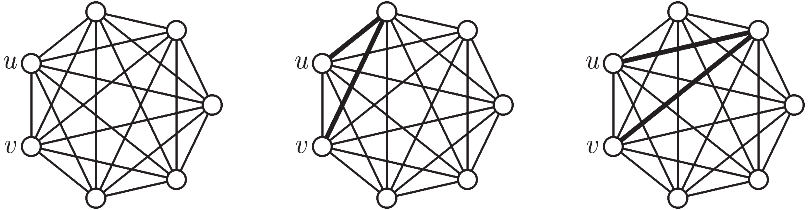
Definice Graf G je *vrcholově k-souvislý*, $k > 1$, pokud i po odebrání libovolných $k-1$ vrcholů z G zůstane výsledný podgraf souvislý nebo K_1.
(úplný graf K_n je z definice vrcholově $(n-1)$-souvislý)



Obrázek: *Grafy s různými stupni souvislosti.*

Věta 7.5. (Mengerovy věty)
Graf G je hranově k-souvislý právě když mezi libovolnými dvěma vrcholy existuje alespoň k hranově-disjunktních cest (vrcholy mohou být sdílené).
Graf G je vrcholově k-souvislý právě když mezi libovolnými dvěma vrcholy existuje alespoň k disjunktních cest (různých až na ty dva spojované vrcholy).

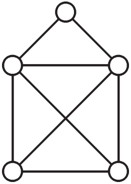
Důkaz V předmětu Teorie Grafů I. □



Obrázek: *Dvě různé disjunktní cesty mezi vrcholy* u, v *v grafu* K_7.

7.4. Eulerovské grafy

Historicky první problém vyřešený pomocí Teorie grafů 1736:
Problém sedmi mostů města Královce – hledání uzavřeného tahu uv, kde $u = v$ a tah obsahuje všechny hrany grafu G.



Obrázek: *Lze tento graf (hrany) nakreslit jedním tahem?*

Věta 7.6. Graf G lze nakreslit jedním uzavřeným tahem, právě když G je souvislý a všechny vrcholy v G jsou *sudého stupně*.

Důsledek 7.7. Graf G lze nakreslit jedním otevřeným tahem právě když G je souvislý a *právě dva* vrcholy v G jsou lichého stupně.

Věta 7.6. Graf G lze nakreslit jedním uzavřeným tahem, právě když G je souvislý a všechny vrcholy v G jsou *sudého stupně*.

Důkaz
Indukcí vzhledem k počtu hran (jen myšlenka důkazu).

Základ indukce:
Lze použít triviální graf. Pro netriviální souvislý graf G je každý vrchol stupně alespoň 2. Pro nejmenší takový graf je $G = C_n$. G je jistě možno nakreslit jedním uzavřeným tahem (proč?).

Indukční krok:
Předpokládejme, že každý souvislý graf s méně než $|E|$ hranami je možno nakreslit jedním uzavřeným tahem. V G najdeme cyklus C (každý vrchol stupně alespoň 2). V grafu $G - C$ jsou vrcholy sudého stupně a nebo izolované vrcholy. Pokud $G - C$ není souvislý, lze každou jeho komponentu dle indukčního předpokladu nakreslit jedním tahem.
Nyní přidáme do cyklu C uzavřený tah pro každý vrchol další v_i, který leží v některé komponentě. Získáme uzavřený tah grafem G.
Podle principu matematické indukce je důkaz hotov. $\square$

Příště

Kapitola 8. Vzdálenost a metrika

- motivace
- vzdálenost v grafu
- výpočet metriky
- vážená (ohodnocená) vzdálenost
- hledání nejkratší cesty