Přehled přednášky

Kapitola 9. Stromy a les

- motivace
- základní vlastnosti stromů
- kořenové stromy
- isomorfizmus stromů
- kostry grafů

Opakování

Budeme používat následující pojmy:

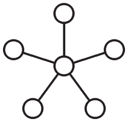
- stupeň vrcholu
- cesta v grafu
- podgraf
- cyklus (kružnice)
- souvislý a nesouvislý graf

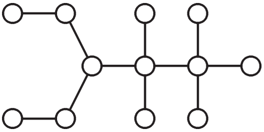
Kapitola 9. Stromy a lesy

Motivace

Jedním z nejběžnějších útvarů v přírodě i v matematice jsou stromy (objekty se „stromovou“ strukturou).
Existuje celá řada motivací, které můžeme popsat „stromem“.

- rodokmeny
- evoluční strom
- elektrické rozvody
- hierarchické struktury (šéf a podřízený)
- větvení při vyhledávání



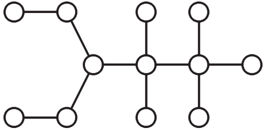


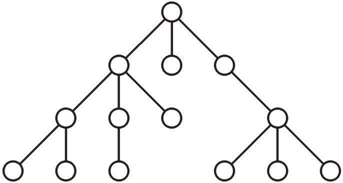
9.1. Základní vlastnosti stromů

Řekneme, že graf neobsahuje cyklus (kružnici), jestliže žádný jeho podgraf není isomorfní cyklu.

Definice 9.1. *Strom* je (konečný) jednoduchý souvislý graf neobsahující cyklus.

Definice *Les* je graf, jehož komponenty jsou stromy.





Poznámka k názvosloví

- *Les* je jednoduchý graf neobsahující cyklus.
- *Strom* je souvislý les.

Což nezní tak pěkně...

Lemma 9.2. Strom s více než jedním vrcholem má nějaký vrchol stupně 1.

Důkaz

Souvislý graf s více než jedním vrcholem nemůže mít vrchol stupně 0. Vezmeme strom T a v něm libovolný vrchol v. Sestrojíme nyní co nejdelší sled S v T začínající ve v. S začneme libovolnou hranou vycházející z v (jistě existuje, proč?). V každém dalším vrcholu u sledu S

- buď u má stupeň alespoň 2 a můžeme prodloužit sled S o další hranu (všimněte si: pokud by se ve sledu S zopakoval některý vrchol, obsahoval by S cyklus, který by byla podgrafem stromu T, což odporuje definici stromu.)

- nebo u sled končí (u je hledaný vrchol stupně 1),

Protože T je konečný, jistě takový vrchol stupně 1 ve stromu T najdeme. □

Poznámka

Není těžké ukázat, že v každém stromu T existují alespoň 2 vrcholy stupně 1.

Otázky:

- Jak se jmenuje strom, který má právě 2 vrcholy stupně 2?
- Existuje strom, který má vrchol stupně k a přitom má méně než k vrcholů stupně 1?
- Umíte Vaše tvrzení z předchozího bodu dokázat?
- Kolik musíme vynechat hran z grafu K_n, abychom dostali strom?

Věta 9.3. Strom na n vrcholech má právě $n-1$ hran pro $n \geq 1$.

Důkaz
Dokážeme indukcí podle n. Strom s jedním vrcholem má $n-1=0$ hran. Nechť T je strom na $n>1$ vrcholech. Podle Lemmatu 9.2. má T vrchol v stupně 1. Označme $T' = T - v$ graf, který vznikne ze stromu T odebráním vrcholu v („oholením“). Graf T' je také souvislý a bez cyklů, proto je T' strom na $n-1$ vrcholech. Podle indukčního předpokladu má T' $(n-1)-1$ hran. Původní strom T má o jednu hranu více, tj $(n-1)-1+1=n-1$ hran. □

Poznámka o důkazu vět ve tvaru $A \Rightarrow B$

kde A je *předpoklad* věty a B je *tvrzení* věty.

Důkaz přímý spočívá v řadě platných implikací.

$$A = A_0 \Rightarrow A_1 \Rightarrow A_2 \Rightarrow \cdots \Rightarrow A_n = B$$

Důkaz nepřímý spočívá v přímém důkazy věty $\neg B \Rightarrow \neg A$, který má stejnou tabulku pravdivostních hodnot jako původní výrok $A \Rightarrow B$.

$$\neg B = A_0 \Rightarrow A_1 \Rightarrow A_2 \Rightarrow \cdots \Rightarrow A_n = \neg A$$

Důkaz sporem je modifikací nepřímého důkazu, kdy předpokládáme, že platí současně předpoklad A a negace tvrzení $\neg B$. Řadou platných implikací dojdeme k tzv. *sporu*. Sporem rozumíme situaci, kdy by měl platit výrok V a současně také jeho negace $\neg V$, což není možné.

$$A = A_0 \Rightarrow A_1 \Rightarrow A_2 \Rightarrow \cdots \Rightarrow A_n = V$$

$$\neg B = A_0 \Rightarrow A_1 \Rightarrow A_2 \Rightarrow \cdots \Rightarrow A_n = \neg V$$

Věta 9.4. Mezi každými dvěma vrcholy stromu existuje právě jedna cesta.

Důkaz Dokážeme sporem.
Budeme vycházet z platnosti předpokladu (T je strom) a negace tvrzení (mezi některými vrcholy stromu T existuje více než jedna cesta).
Protože strom T je souvislý (podle definice stromu), vede mezi libovolnými dvěma vrcholy u, v nějaká cesta. Pokud mezi u, v existují dvě různé cesty, tak jejich sjednocení je uzavřeným sledem v T a po „vynechání“ všech opakujících se vrcholů v tohoto sledu vznikne cyklus v T, což je spor s předpokladem.
Proto mezi u a v existuje jen jedna cesta. □

Důsledek 9.5. Přidáním jedné (nové) hrany do stromu vznikne graf s právě jedním cyklem.

Důkaz
Nechť mezi vrcholy u, v ve stromu T není hrana. Přidáním hrany $e = uv$ vznikne právě jeden cyklus spojením e a jediné cesty mezi vrcholy u, v v T (jediné podle Věty 9.4.). □

Máme dán nějaký souvislý graf G. Při určování stupně hranové souvislosti jsme se ptali, kolik *nejméně* stačí z grafu odstranit hran, aby se G stal nesouvislý. Má smysl se ptát,

- kolik *nejvíce* hran můžeme z grafu G odstranit, aby G zůstal souvislý, nebo naopak
- kolik *nejméně* hran musí v grafu G zůstat, aby byl souvislý

Právě stromy jsou grafy, které jsou souvislé a už z nich nemůžeme odebrat žádnou hranu, aniž by se porušila souvislost.

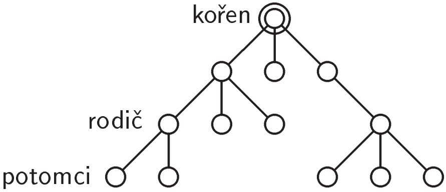
Věta 9.6. Strom je *minimální souvislý* graf (na daných vrcholech).

Důkaz Strom je souvislý podle definice. Pokud souvislý graf obsahuje cyklus, zůstane souvislý i po vynechání některé hrany kružnice. Proto je minimální souvislý graf stromem.
Naopak, pokud by vynecháním hrany $e = uv$ ze stromu T vznikl souvislý graf, pak by mezi vrcholy u, v v T existovaly dvě cesty: cesta z u do v v $T \setminus e$ a hrana $e = uv$. To by byl spor s Větou 9.4.
Proto je strom minimálním souvislým grafem na daných vrcholech. □

9.2. Kořenové stromy

Někdy je užitečné při práci se strukturou typu „strom" označit význačný vrchol, tzv. *kořen*, („začátek" uložených dat). Kořenové stromy mají tradiční motivaci i v rodokmenech („O Foxíkovi" z Dášenky) odtud vychází i terminologie.

Definice 9.8. *Kořenovým stromem* je strom T spolu s vyznačeným *kořenem* $r \in V(T)$, zkráceně zapisujeme dvojicí T, r (případně (T, r)).



Obrázek: *Kořenový strom, rodič a potomek.*

Definice Mějme kořenový strom T, r a v něm dvojici vrcholů u, v, kde u je sousední s v na cestě ke kořenu r. Pak u nazýváme *rodičem* vrcholu v a v nazýváme *potomkem* vrcholu u.

Definice

Vrchol stupně 1 v libovolném stromu nazýváme *listem* nebo také *koncovým vrcholem*.

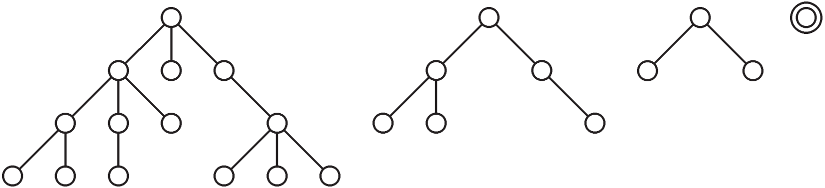
Kořen může být listem, avšak i v tom případě preferujeme název *kořen*.
V kořenovém stromu listem rozumíme vrcholy, které nemají potomky a nejsou kořenem.

Definice

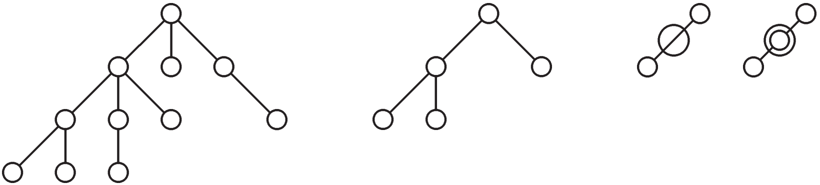
Centrem stromu T rozumíme vrchol nebo hranu v T určenou následujícím postupem:

- Pokud má strom T jediný vrchol v, je v centrum stromu T.
  Pokud má strom T dva vrcholy, je jeho centrem hrana spojující tyto dva vrcholy.
- Jinak vytvoříme (menší) strom $T' \subset T$ vynecháním všech listů T (oholením). Je zřejmé, že T' je neprázdný; vracíme se na předchozí bod. Rekurzí získané centrum T' bude zároveň centrem T.

Příklad:



Obrázek: *Hledání centra.*

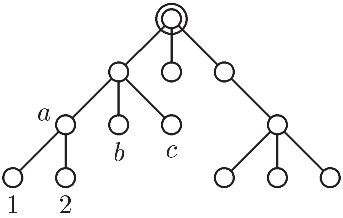


Obrázek: *Hledání centra.*

Budeme-li chtít nějakému stromu přiřadit kořen *jednoznačně*, je nejlepší jej přiřadit *centru* stromu (protože centrum je určeno jednoznačně).
V případě, že centrem stromu je hrana, bude kořenem nový vrchol „rozdělující“ tuto hranu na dvě.

Další informací vázanou ke kořenovým stromům bývá uspořádání potomků každého vrcholu (jako seřazení potomků jedné generace v rodokmenech podle jejich data narození).

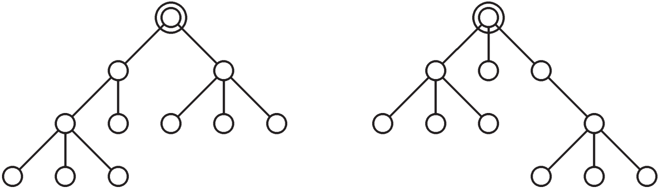
Definice Kořenový strom T, r je *uspořádaný*, jestliže pro každý jeho vrchol je jednoznačně dáno pořadí jeho potomků („zleva doprava“).
Uspořádaný kořenový strom se také nazývá *pěstovaný strom*.



9.3. Isomorfizmus stromů

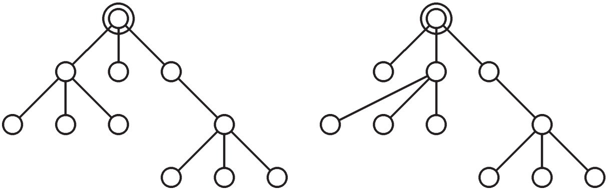
Pojem *isomorfismus stromů* je speciálním případem isomorfismu grafů. Dva stromy jsou isomorfní, pokud jsou isomorfní jako grafy.
Připomeňme, že pro úlohu rozhodnout, zda dva grafy jsou isomorfní není znám žádný rychlý algoritmus. Stromy jsou natolik speciální třída grafů, že pro ně takový algoritmus existuje!
Než si ho ukážeme, zavedeme několik pojmů.

Definice Dva kořenové stromy T, r a T', r' jsou *isomorfní* pokud existuje isomorfismus mezi stromy T a T', který zobrazí *kořen* r *na kořen* r'.



Obrázek: *Isomorfní stromy, které nejsou isomorfní jako kořenové stromy.*

Definice Dva uspořádané kořenové (pěstované) stromy jsou isomorfní, jestliže pro ně existuje isomorfismus kořenových stromů, který navíc *zachová pořadí potomků* všech vrcholů.

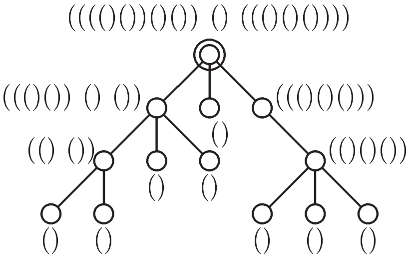


Obrázek: *Isomorfní kořenové stromy, které nejsou isomorfní jako pěstované.*

Kódování uspořádaných kořenových stromů

Uspořádanému kořenovému stromu lze snadným postupem přiřadit řetězec vnořených závorek, který jej jednoznačně popisuje.

Definice *Kód* uspořádaného kořenového stromu se sestaví rekurzivně z kódů všech podstromů kořene, seřazených v daném (pevně zvoleném) pořadí a uzavřených do páru závorek (viz obrázek).



Obrázek: *Kódování pěstovaného stromu.*

Poznámka Místo znaků „(“ a „)“ můžeme použít jiné symboly, třeba 0 a 1.

Lemma 9.10. Dva uspořádané kořenové (pěstované) stromy jsou isomorfní právě když jejich kódy získané podle předchozího popisu jsou shodné řetězce.

Fakt Máme dán kód s uspořádaného kořenového stromu, příslušný strom nakreslíme následujícím postupem:

- Při přečtení znaku „(“ na začátku spustíme pero na papír, do kořene.
- Při každém dalším přečtení znaku „()“ nakreslíme hranu do následujícího potomka současného vrcholu.
- Při každém přečtení znaku „)“ se perem vrátíme do rodiče současného vrcholu, případně zvedneme pero, pokud jsme v kořenu.

Při určování isomorfismu obecných stromů použijeme

- Lemma 9.10. pro jednoznačnou reprezentaci závorkovým kódem.
- Kořen volíme v centru, potomky seřadíme vzestupně lexikograficky podle jejich kódů.

Dostaneme algoritmus, který je popsán dále.

Algoritmus 9.12. Určení isomorfismu stromů.

Algoritmus zjišťuje, zda dané dva stromy T a U jsou isomorfní ($T \simeq^{?} U$)

```
Vstup < stromy T a U;
for (X=T,U) {
    // určení center daných stromů U, T
    x = centrum(X);
    if (x je jeden vrchol) r = x;
    else nový vrchol r, nahraď hranu x=uv hranami ru, rv;
    k[X] = minimalni_kod(X,r);
}
if (k[T]==k[U] jako řetězce)
    vypiš("Jsou isomorfní.");
else
    vypiš("Nejsou isomorfní.");
exit;
```

Algoritmus využívá funkci `minimalni_kod(X,r)`, která pro strom X s kořenem r najde (lexikograficky) minimální kód.

Algoritmus 9.12. ... pokračování

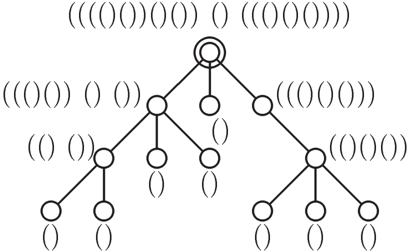
Funkce `minimalni_kod(X,r)` najde pro strom X s kořenem r (lexikograficky) minimální kód.

```
funkce  minimalni_kod(strom X, vrchol r) {
   if (X má jeden vrchol)  return "()";
   Y[1...d] = {souvislé komponenty X-r, podstromy kořene bez r}
   s[1...d] = {kořeny podstromů Y[] v odpovídajícím pořadí};
   for (i=1,...,d)
       k[i] = minimalni_kod(Y[i],s[i]);
   sort lexikograficky podle klíče k[1] <= k[2] <= ... <= k[d];
   return "("+k[1]+...+k[d]+")";
}
```

Funkce hledá minimální kód rekurzivně.

Otázky:

- Je následující kód minimální? Proč?
- Jak vypadá minimální kód?



Obrázek: *Kód pěstovaného stromu.*

Poznámka

Pro lexikogafické uspořádání může být přehlednější použít označení pomocí „0“ a „1“ místo „(“ a „)“.

Příště

Kapitola 9. Stromy a les

- kostry grafů

Bonus

- Hamiltonovské grafy

Kapitola 10. Barevnost a kreslení grafů

- motivace
- barevnost grafů
- rovinné nakreslení grafů
- rozpoznání rovinných grafů
- barvení map a rovinných grafů