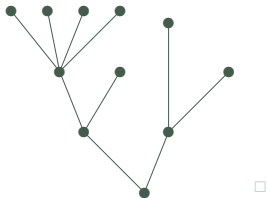


4 Stromy a les

Jedním ze základních, a patrně nejjednodušším, typem grafů jsou takzvané *stromy*. Jedná se o souvislé grafy bez kružnic. Přes svou (zdánlivou) jednoduchost mají stromy bohatou strukturu a především množství vlastních aplikací, například v datových strukturách.



Patrně nejstarší motivací pojmu stromu jsou rodokmeny (“po meči”), jejichž původ sahá daleko před vznik teorie grafů.

Stručný přehled lekce

- Definice a základní vlastnosti stromů.
- Kořenové a uspořádané stromy, isomorfismus.
- Kostry grafů a jejich počet.

4.1 Základní vlastnosti stromů

Definice 4.1. **Strom** je jednoduchý **souvislý** graf T **bez kružnic**.

Lema 4.2. *Strom s více než jedním vrcholem obsahuje vrchol **stupně 1**.* \square

Důkaz. Souvislý graf s více než jedním vrcholem nemůže mít vrchol stupně 0. Proto vezmeme strom T a v něm libovolný vrchol v . Sestrojíme nyní co nejdelší sled S v T začínající ve v :

S začne libovolnou hranou vycházející z v . V každém dalším vrchole u , do kterého se dostaneme a má stupeň větší než 1, lze pak pokračovat sled S další novou hranou. Uvědomme si, že pokud by se ve sledu S poprvé zopakoval některý vrchol, získali bychom kružnici, což ve stromě nelze. Proto sled S musí jednou skončit v nějakém vrcholu stupně 1 v T . \square

Věta 4.3. *Strom na n vrcholech má přesně $n - 1$ hran pro $n \geq 1$.* \square

Důkaz. Toto tvrzení dokážeme indukcí podle n .

Strom s jedním vrcholem má $n - 1 = 0$ hran.

Nechť T je strom na $n > 1$ vrcholech. Podle Lematu 4.2 má T vrchol v stupně 1. Označme $T' = T - v$ graf vzniklý z T odebráním vrcholu v . Pak T' je také souvislý bez kružnic, tudíž strom na $n - 1$ vrcholech. Dle indukčního předpokladu T' má $n - 1 - 1$ hran, a proto T má $n - 1 - 1 + 1 = n - 1$ hran. \square

Věta 4.4. Mezi každými dvěma vrcholy stromu vede právě *jediná cesta*. □

Důkaz. Jelikož strom T je souvislý dle definice, mezi libovolnými dvěma vrcholy u, v vede nějaká cesta. Pokud by existovaly dvě různé cesty P_1, P_2 mezi u, v , tak bychom vzali jejich symetrický rozdíl, podgraf $H = P_1 \Delta P_2$ s neprázdnou množinou hran, kde H zřejmě má všechny stupně sudé. Na druhou stranu se však podgraf stromu musí opět skládat z komponent stromů, a tudíž obsahovat vrchol stupně 1 podle Lematu 4.2, spor. Proto cesta mezi u a v existuje jen jedna. □

Důsledek 4.5. Přidáním jedné nové hrany do stromu vznikne právě jedna kružnice.

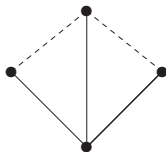
Důkaz. Nechť mezi vrcholy u, v ve stromu T není hrana. Přidáním hrany $e = uv$ vznikne právě jedna kružnice z e a jediné cesty mezi u, v v T podle Věty 4.4. □ □

Věta 4.6. Strom je *minimální souvislý graf* (na daných vrcholech). □

Důkaz. Strom je souvislý podle definice. Pokud by souvislý graf měl kružnici, zůstal by souvislý i po vypuštění některé hrany té kružnice. Proto minimální souvislý graf (na daných vrcholech) je stromem. Naopak, pokud by vypuštěním hrany $e = uv$ ze stromu T vznikl souvislý graf, pak by mezi u, v v T existovaly dvě cesty (dohromady kružnice) – hrana e a jiná cesta v $T - e$. To je ve sporu s Větou 4.4. Proto je strom minimálním souvislým grafem na daných vrcholech. □

Příklad 4.7. Kolik nejvýše kružnic vznikne v grafu, který vytvoříme ze stromu přidáním dvou hran? □

Přidáním jedné hrany do stromu T vznikne jedna kružnice dle Důsledku 4.5. Druhá hrana vytvoří nejméně ještě jednu kružnici ze stejných důvodů, ale může vytvořit i dvě další kružnice, jako třeba v následujícím grafu, kde strom T je vyznačen plnými čarami a dvě přidané hrany čárkovaně.



Každá z přidaných dvou hran vytvoří vlastní trojúhelník a navíc ještě vznikne kružnice délky 4 procházející oběma z přidaných hran.

Na druhou stranu chceme ukázat, že více než 3 kružnice vzniknout nemohou po přidání dvou hran e, f do stromu T : Podle Důsledku 4.5 vznikne jen jedna kružnice procházející hranou e a neobsahující f , stejně tak jedna kružnice procházející f a neobsahující e . Nakonec stačí nahlédnout, že je nejvýše jedna možná kružnice procházející oběma hranami e, f : Pokud by takové byly dvě různé C_1, C_2 , podívali bychom se na jejich symetrický rozdíl, podgraf $H = C_1 \Delta C_2$, který má všechny stupně sudé, neprázdnou množinu hran a je navíc pografem stromu T . Takže stejně jako ve Větě 4.4 dostáváme spor s faktem, že podgrafy stromů s hranami musí obsahovat vrchol stupně 1. □

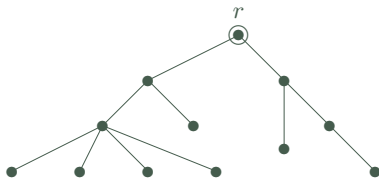
4.2 Kořenové stromy

Při mnoha aplikacích stromových struktur se ke stromu jako grafu samotnému ještě váží dodatečné informace, jako třeba vyznačený jeden vrchol, tzv. kořen stromu, ze kterého celý strom “vyrůstá”. Typickým příkladem jsou různé (acyklické) datové struktury, ve kterých je vyznačený vrchol – kořen, referován jako “začátek” uložených dat. □

Kořenové stromy mají také tradiční motivaci v rodokmenech a z toho vychází jejich běžná terminologie.

Definice 4.8. Kořenovým stromem je strom T spolu s vyznačeným *kořenem* $r \in V(T)$, zkráceně dvojice T, r . □

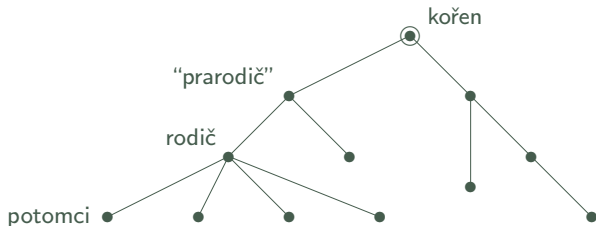
Příklad kořenového stromu je na následujícím obrázku:



Zajímavostí je, že v informatice stromy většinou rostou od kořene směrem dolů. (Však také nejsme v biologii. . .)

Definice: Mějme kořenový strom T , r a v něm vrchol v . Označme u souseda v na cestě směrem ke kořeni r . Pak je u nazýván *rodičem* v a v je nazýván *potomkem* u . □

Kořen nemá žádného rodiče.



Často se také setkáte v kořenových stromech s označováním "otec–syn" místo rodič–potomek. My jsme takové označení nepoužili proto, že by (hlavně v zemích na západ od nás) mohlo být považováno za "sexistické". □

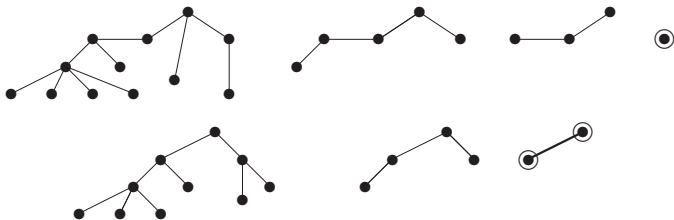
Definice: Vrchol stupně 1 v libovolném stromu nazýváme *listem*.

Pozor, i kořen stromu může být listem, pokud má stupeň 1, ale obvykle se to tak neříká. List kořenového stromu, který není kořenem, nemá potomky.

Definice: *Centrem stromu* T rozumíme buď vrchol nebo hranu nalezenou v T následujícím postupem:

- Pokud má strom T jeden vrchol, je to jeho centrum. Pokud má strom T dva vrcholy, je jeho centrem hrana spojující tyto dva vrcholy. □
- Jinak vytvoříme menší strom $T' \subset T$ vypuštěním všech listů T najednou. Je zřejmé, že T' je neprázdný, a vracíme se na předchozí bod. Získané (rekurzivně) centrum T' je zároveň centrem T . □

Příklad 4.9. *Ilustrací definice centra jsou následující dva postupy nalezení:*



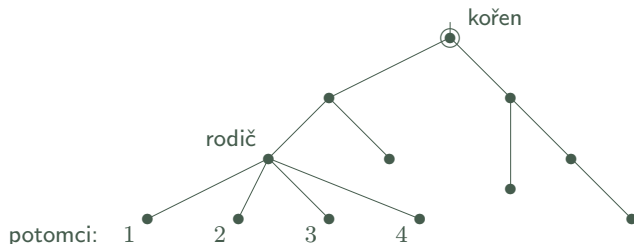
□

Fakt. Pokud chceme danému (abstraktnímu) stromu přiřadit jednoznačně kořen, je nejlepší jej přiřadit centru stromu. Speciálně, pokud je centrem hrana, bude kořenem nový vrchol “rozdělující” tuto hranu na dvě.

Definice: Kořenový strom T, r je *uspořádaný*, pokud je pro každý jeho vrchol jednoznačně dáno pořadí jeho potomků (“zleva doprava”).

Uspořádaný kořenový strom se také nazývá *pěstovaný strom*. □

Uspořádaný kořenový strom si jinak také můžeme představit jako strom s vyznačeným kořenem a pevně zvoleným nakreslením v rovině bez křížení hran. Nakreslení hran potomků vzhledem k hraně rodiče pak udává (ve zvolené orientaci) pořadí potomků.



Uspořádání potomků vrcholu ve stromu je přirozeně požadováno v mnoha praktických situacích. Například ve stromových datových strukturách jsou často potomci explicitně seřazeni podle daného klíče, jako třeba ve vyhledávacích binárních stromech. I v případech, kdy uspořádání potomků ve stromě není dáno, je možné jej jednoznačně definovat, jak uvidíme v následující části.

4.3 Isomorfismus stromů

Jelikož stromy jsou speciálním případem grafů, je isomorfismus stromů totéž co isomorfismus grafů obecně. Avšak na rozdíl od grafů, kdy je určení isomorfismu těžký problém, pro isomorfismus stromů existuje efektivní postup, který si ukážeme. □

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

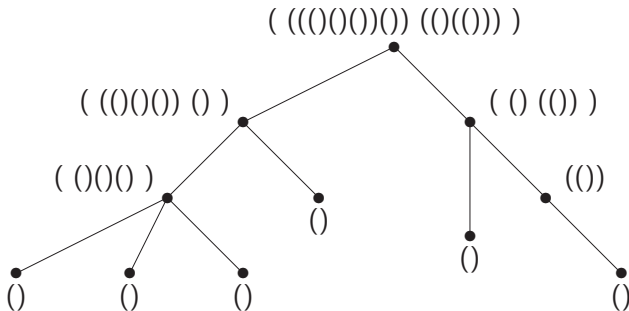


Definice: Dva uspořádané kořenové stromy jsou isomorfní pokud je mezi nimi isomorfismus kořenových stromů, který navíc **zachovává pořadí potomků** všech vrcholů.



Kódování uspořádaných kořenových stromů ("zavorkování" stromů)

Definice:



Kód uspořádaného kořenového stromu se počítá rekurzivně z kódů všech podstromů kořene, seřazených v daném pořadí a uzavřených do páru závorek. □

Poznámka: Místo znaků '(' a ')' lze použít i jiné symboly, třeba '0' a '1'.

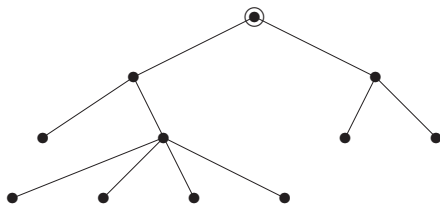
Lema 4.10. *Dva uspořádané kořenové (pěstované) stromy jsou isomorfní právě když jejich kódy získané podle předchozího popisu jsou shodné řetězce.*

Fakt. Je-li 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 už jsme v kořeni. □

Příklad 4.11. Nakreslete jako pěstovaný strom ten odpovídající závorkovému kódu

$((()((()()()()))) (()))$.



□

Při určování isomorfismu obecných stromů použijeme závorkový kód, pro který kořen volíme v centru a potomky seřadíme podle jejich kódů vzestupně lexikograficky.

Algoritmus 4.12. Určení isomorfismu dvou stromů.

```
Vstup < stromy  $T$  a  $U$  ;
for (X=T,U) {
    x = centrum(X);
    if (x je jeden vrchol) r = x;
    else nový r, nahrad' hranu x=uv hranami ru, rv;
    k[X] = minimalni_kod(X,r);
}
if (k[T]==k[U] jako řetězce) printf("Jsou isomorfní.");
else printf("Nejsou isomorfní."); □
exit;

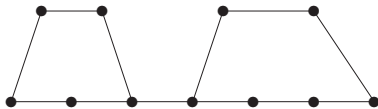
Funkce minimalni_kod(strom X, vrchol r) {
    if (X má jeden vrchol) return "()";
    Y[1...d] = {souvislé komponenty X-r, tj. podstromy kořene r};
    s[1...d] = kořeny podstromů Y[] v odpovídajícím pořadí;
    for (i=1,...,d)
        k[i] = minimalni_kod(Y[i],s[i]);
    sort lexikograficky (abecedně)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

4.4 Kostry grafů

Definice 4.14. **Kostrou** souvislého grafu G

je podgraf v G , který je sám stromem a obsahuje všechny vrcholy grafu G . \square

Příklad 4.15. *Kolik různých koster má tento graf?*



Podívejme se na kostru grafu takto – jaké hrany z grafu vymažeme, aby zbyl strom? Zjistíme musíme vymazat některou hranu z první kružnice (5 možností) a některou hranu z druhé kružnice (6 možností). Na druhou stranu to v tomto jednoduchém příkladě už stačí, vždy pak zbude strom. Výběr vymazané hrany z první kružnice je nezávislý na druhé kružnici (jsou disjunktní), a proto dle principu nezávislých výběrů máme $5 \cdot 6 = 30$ možností vybrat dvě hrany k vymazání. Celkem tedy vyjde 30 koster. \square

Následující výsledek patří ke „krásným drahokamům“ teorie grafů.

Věta 4.16. Úplný graf K_n má přesně n^{n-2} koster. \square

Definice. Laplaceova matice $Q = (q_{ij})_{i,j=1}^n$ grafu G o n vrcholech je definována:

- $q_{ii} = d_G(i)$ (stupeň vrcholu),
- $q_{ij} = 0$ pro vrcholy $i \neq j$ nespojené hranou,
- $q_{ij} = -1$ pro vrcholy $i \neq j$ spojené hranou. \square

Věta 4.17. Nechť Q je Laplaceova matice grafu G a matice Q' vznikne vyškrtnutím jejího prvního řádku a sloupce. Pak počet koster grafu G je roven determinantu $|Q'|$.

Důkaz této překvapivé věty je mimo dosah naší přednášky (využívá silné nástroje lineární algebry).

Uvědomte si, proč samotná matice Q je singulární (determinantu 0) – neboť součet prvků v každém řádku je 0.

Je také možno vyškrtávat jiné řádky a sloupce, ale může se tím změnit znaménko determinantu.