

PB165 – Grafy a sítě

Grafy a algoritmy v komunikačních sítích

Průsvitky

- průběžná aktualizace v IS Studijní materiály

Odpovědníky v IS

- příklady na procvičení k jednotlivým přednáškám
- procvičení i dalších základních grafových témat, které byste měli znát

Hodnocení

- vnitrosemestrální písemná práce
 - body započítány 20% do výsledné známky
 - termín: na páté přednášce 26.10.
 - 50 minut, cca 5 příkladů z obsahu proběhlých přednášek
- závěrečná písemná práce
 - body započítány 80% do výsledné známky
- podle dosažených procent:
A $\langle 100, 89 \rangle$, B $\langle 89, 79 \rangle$, C $\langle 79, 69 \rangle$, D $\langle 69, 59 \rangle$, E $\langle 59, 55 \rangle$

- Návaznost na: IB000 Matematické základy informatiky
IB002 Algoritmy a datové struktury I
 - zopakujte si studijní materiály IB000 a IB002 ke grafům
 - vybrané hlavní koncepty stručně zopakujeme
- ① Síťové grafy
 - 2 přednášky, Rudová & Hladká
- ② Plánování a rozvrhování na síťových grafech
 - 4 přednášky, Rudová
- ③ Modelování sítí
 - Hladká & Matyska

Síťové grafy

- 1 Typy grafů v počítačových sítích
 - 2 Směrování - hledání nejkratších cest
- ### Plánování a rozvrhování na síťových grafech
- 3 Úvod k plánování a rozvrhování
 - 4 Plánování projektu, plánování s barvením grafu
 - 5 ... písemka ...
 - 6 Plánování na počítačové síti I
 - 7 Plánování na počítačové síti II

Modelování sítí

- 8 Zajímavé grafy a peer to peer sítě
- 9 Topologie internetu
- 10 Toky a kódování v síti
- 11 Heuristiky a plánování v telekomunikačních sítích
- 12 Využití grafů při analýze sociálních sítí

- 1 Úvod do technik diskrétní matematiky pro podporu návrhu a řízení komunikačních sítí
- 2 Grafově teoretický koncept
- 3 Ukázky aplikací v komunikačních sítích

- Teorie grafů je důkladně rozpracována a nabízí mnoho využitelných algoritmů.
- Graf představuje velmi přímočarou reprezentaci sítě na všech úrovních OSI modelu, např.:
 - Síťové prvky a jejich fyzické propojení na nejnižší vrstvě,
 - síťové aplikace a TCP spojení mezi nimi na transportní vrstvě,
 - procesy distribuovaného výpočtu a jejich komunikace na aplikační vrstvě.
- Grafy nacházejí využití i v návrhu síťových protokolů a jejich formální verifikaci.

- Diskrétní struktury
 - grafy
 - hypergrafy
 - kombinatorika
- Algoritmy
 - procedurální popis „krok za krokem“ problémů, které nelze řešit bezprostředně
 - složitost, NP - těžké problémy
- Matematická optimalizace
 - vývoj a implementace pro podporu rozhodování
 - problém návrhu sítí, identifikace úzkých míst
 - obchodní aspekty sítí
 - modeluje se grafy
- Distribuované výpočty
 - síť jako distribuovaný systém
 - příklad směrovací algoritmy, P2P sítě

Obsah: Typy grafů používané v počítačových sítích

- Graf, orientovaný graf, ohodnocený graf opakování
 - Multigraf
 - Strom opakování
 - Kořenový strom
 - n-ární strom, uspořádaný strom
 - Binární strom
- +
- Aplikace grafů v počítačových sítích
 - Implementace grafů

Nejjednodušší diskretní strukturou pro modelování síťových problémů jsou grafy. Intuitivně se graf skládá z:

- **Vrcholů (uzlů)**, znázorňovaných schematicky jako „body“,
- **hran** spojujících vrcholy.

Co lze reprezentovat grafem?

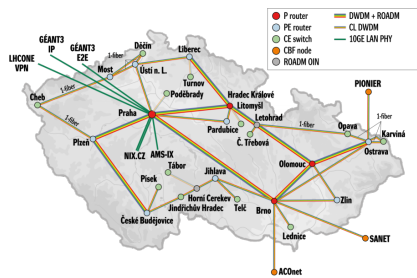
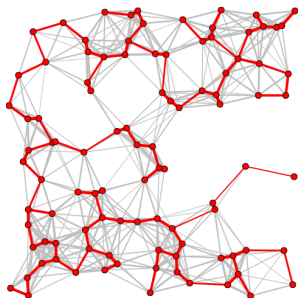
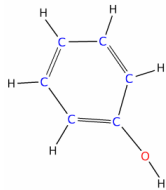
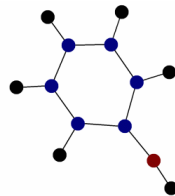
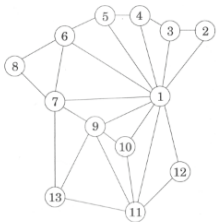
- Mapu měst a silničního spojení,
- atomy v molekule a jejich vazby,
- vodovodní, elektrické sítě

a zejména

- počítačové sítě.

Označován také jako **jednoduchý** graf.

Grafy a sítě - příklady



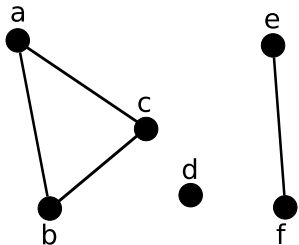
Opakování: definice grafu

Definice

Graf G je uspořádaná dvojice množin (V, E) , kde

- V je množina **vrcholů** a
- E je množina **hran** – množina vybraných **dvouprvkových podmnožin** množiny V

Vrcholy spojené hranou se nazývají **sousední**. Hrana se označuje jako **incidentní** k vrcholům, které spojuje.



$$V = \{a, b, c, d, e, f\}$$

$$E = \left\{ \begin{array}{l} \{a, b\}, \\ \{a, c\}, \\ \{b, c\}, \\ \{e, f\} \end{array} \right\}$$

Hrany v grafu, jak byly definovány, spojují dva rovnocenné vrcholy. Takové grafy se také nazývají **neorientované**. U hran ovšem můžeme vyznačit směr, kterým vedou – hrany i graf se poté nazývají **orientované**.

Definice

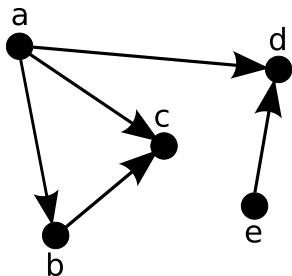
*Graf, jehož hrany jsou uspořádané dvojice vrcholů, se nazývá **orientovaný**.*

O hraně (u, v) říkáme, že **vychází z** vrcholu u a **vstupuje do** vrcholu v . Graficky se orientace hrany označí šipkou ve směru, kterým hrana vede.

Orientovaný graf - příklady

Kde najdeme orientované grafy v sítích?

- Webové stránky – graf odkazů
- DNS – hierarchická struktura domén, serverů
- Směrování – graf cest paketů k cíli



$$V = \{a, b, c, d, e\}$$
$$E = \{ (a, b), (a, c), (b, c), (a, d), (e, d) \}$$

Definice grafu povoluje nejvýše 1 hranu mezi každou dvojicí vrcholů a požaduje, aby hrana spojovala různé vrcholy. Tato omezení odstraňuje multigraf:

Definice

***Multigraf** je graf, jenž nahrazuje množinu hran **multimnožinou** (smí obsahovat násobné prvky) a umožňuje existenci **smyček** – hran spojujících vrchol sám ze sebou.*

Multigraf lépe odpovídá reálným fyzickým sítím, kde se často vyskytují redundantní linky.

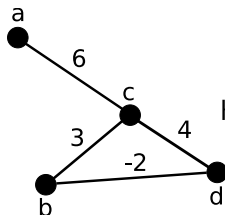
Smyčky mohou znázornit např. loopback – rozhraní přijímající zprávy, které samo vysílá.

Opakování: Ohodnocení grafu

Vrcholům a hranám je možné přiřadit např. číslo či barvu.

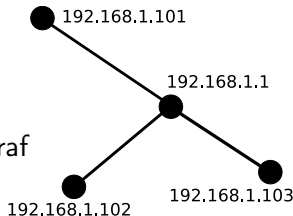
Definice

Přiřazení prvků konečné množiny vrcholům či hranám grafu nazýváme jejich **ohodnocením**.



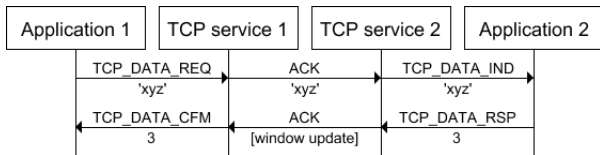
Hranově ohodnocený graf

Vrcholově ohodnocený graf



Příklady ohodnocení na sítích

- Ohodnocení vrcholů:
ohodnocení síťových zařízení L2 a L3 adresami (viz ARP protokol)
- Ohodnocení hran jako ohodnocení linek:
šířka pásma, latence, cena přenosu za jednotku dat
- Ohodnocení vrcholů názvem stavu, hran typem zprávy při abstraktním návrhu protokolů (MSC – Message Sequence Charts)



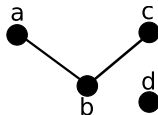
MSC datového přenosu pro TCP

Definice

Stupněm vrcholu v neorientovaném grafu nazýváme počet hran incidentních k vrcholu.

- Počet klientů připojených k Wi-Fi AP
- Počet uzavřených spojení spojovaného protokolu (např. TCP)

Stupeň vrcholu u značíme $deg(u)$.

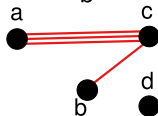


$$deg(a) = 1$$

$$deg(c) = 1$$

$$deg(b) = 2$$

$$deg(d) = 0$$



$$deg(a) = 3$$

$$deg(c) = 4$$

$$deg(b) = 1$$

$$deg(d) = 0$$

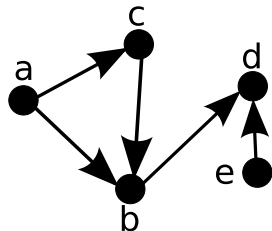
Stupeň vrcholu v orientovaném grafu

V případě orientovaného grafu rozlišujeme **vstupní** a **výstupní** stupeň.

Definice

Vstupním (**výstupním**) stupněm vrcholu u neorientovaného grafu nazýváme počet hran vstupujících do, resp. vycházejících z vrcholu u a značíme jej $\text{deg}^+(u)$, resp. $\text{deg}^-(u)$.

- V grafu odkazů mezi webovými stránkami reprezentuje vstupní stupeň počet odkazů na vedoucí stránku.



vrchol	deg^-	deg^+
a	2	0
b	1	2
c	1	1
d	0	2
e	1	0

Definice

Sledem v grafu (neorientovaném grafu) nazýváme posloupnost vrcholů a hran

$$v_0, e_1, v_1, \dots, e_n, v_n,$$

kde každá hrana e_i spojuje vrcholy v_{i-1}, v_i , resp. vede z v_{i-1} do v_i .

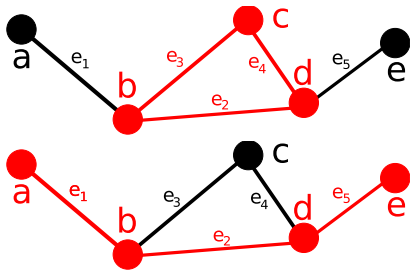
- Sled v grafu je tedy „trasou“, na které se mohou vrcholy i hrany opakovat.
- Samostatný vrchol je také sledem.
- Se sledy se lze setkat i v reálných sítích:
 - Cesta paketu sítí (některé směrovací algoritmy nezabraňují zacyklení paketu v průběhu výpočtu).

Definice

Cesta v grafu je sled bez opakování vrcholů.

V cestě se v důsledku neopakování vrcholů nemohou opakovat ani hrany.

- Cesty definující směrování paketů mezi dvojicemi síťových prvků.



$b, e_2, d, e_3, c, e_4, b$ je sledem v grafu, ale nikoliv cestou – vrchol b se opakuje.

$a, e_1, b, e_2, d, e_5, e$ je sledem i cestou v grafu.

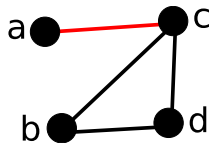
Definice

Souvislý graf je takový (neorientovaný) graf, pokud mezi jeho každými dvěma vrcholy vede cesta.

Definice

Nahradíme-li všechny hrany orientovaného grafu G neorientovanými a získáme-li tak souvislý graf, G je *slabě souvislý*.

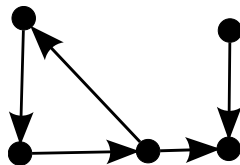
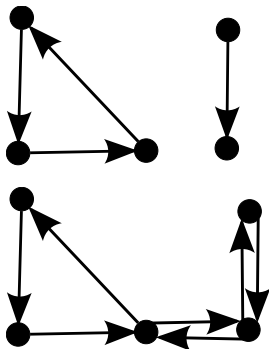
Orientovaný graf je *silně souvislý*, pokud mezi každými dvěma jeho vrcholy vedou cesty v obou směrech.



Tento graf je souvislý; po odebrání vyznačené hrany by souvislý nebyl, odebrání jedné z nevyznačených hran by jeho souvislost zachovalo.

Souvislost grafu – příklady

- Internet na fyzické vrstvě tvoří souvislý graf.
- Internet na IP vrstvě tvoří slabě souvislý (ovšem silně nesouvislý) orientovaný graf (adresy za NAT).
- Orientovaný graf webových stránek není silně ani slabě souvislý.



Grafy na obrázcích jsou:

- 1 Nesouvislý
- 2 Slabě souvislý
- 3 Silně souvislý

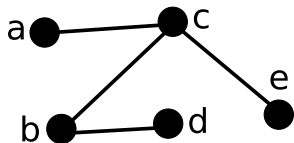
Jak efektivně uložit graf v paměti počítače či aktivního síťového prvku? Vrcholy označíme čísly $1 \dots n$. Pro uložení hran máme dvě základní možnosti:

- matice sousednosti,
- seznamy sousedů.

Matice sousednosti

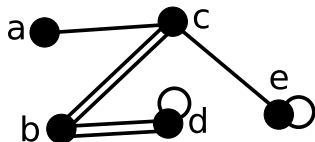
- matice E o rozměrech $n \times n$ pro n vrcholů grafu
- $E_{ij} = 1$ pokud hrana (i, j) patří do grafu
- $E_{ij} = 0$ jinak
- pro neorientovaný graf je symetrická
pro orientovaný graf nemusí být symetrická

Matice sousednosti



	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>
<i>a</i>	0	0	1	0	0
<i>b</i>	0	0	1	1	0
<i>c</i>	1	1	0	0	1
<i>d</i>	0	1	0	0	0
<i>e</i>	0	0	1	0	0

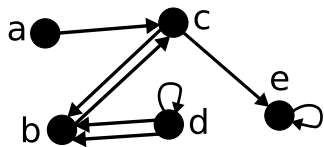
V této podobě je matice sousednosti vhodná jen pro reprezentaci jednoduchého grafu. Multigraf lze reprezentovat maticí sousednosti, jejíž prvky udávají počet hran mezi každými dvěma vrcholy:



	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>
<i>a</i>	0	0	1	0	0
<i>b</i>	0	0	2	2	0
<i>c</i>	1	2	0	0	1
<i>d</i>	0	2	0	1	0
<i>e</i>	0	0	1	0	1

Obdobně lze ukládat jednoduchý hranově ohodnocený graf.

Matice sousednosti pro orientovaný multigraf

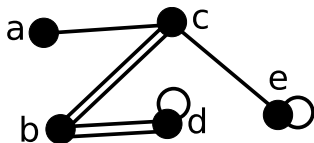


	a	b	c	d	e
a	0	0	1	0	0
b	0	0	1	0	0
c	0	1	0	0	1
d	0	2	0	1	0
e	0	0	0	0	1

Pro každý vrchol existuje samostatný seznam sousedů, s nimiž je vrchol spojen hranou (či do nich vede orientovaná hrana).

- Lze implementovat jako:
 - seznam sousedů pomocí ukazatelů
 - pole velikosti $|V|$ pro jednotlivé vrcholy
 - od každého prvku pole (vrcholu i):
seznam provázaný ukazateli odkazující na sousedy vrcholu i
 - seznam sousedů pomocí 2 jednorozměrných polí:
 - V jednom jsou uloženy všechny seznamy za sebou, seřazené podle čísla vrcholu
 - Druhé uchovává indexy, na kterých začínají v prvním poli sousedé každého vrcholu
- Násobné hrany v multigrafu jsou zadány násobným uvedením vrcholu v seznamu sousedů

Seznamy sousedů – příklad



Pole indexů do seznamu sousedů:

a	b	c	d	e
1	2	6	10	13

Seznam sousedů:

1	2	3	4	5	6	7	8	9	10	11	12	13	14
c	c	c	d	d	a	b	b	e	b	b	d	c	e

- Reprezentace: prostřednictvím dvou polí vs. seznamů provázaných ukazateli **pro počítačové sítě?**
- Reprezentace prostřednictvím dvou polí je pro počítačové sítě výhodnější než reprezentace pomocí seznamů provázaných ukazateli.
 - U počítačových sítí nedochází k častým změnám struktury grafu.
 - pro „řidké“ grafy (výrazně méně než n^2 hran) jsou seznamy sousedů výhodnější z hlediska paměťové náročnosti než matice sousednosti. Takových grafů je mezi sítěmi většina.

Definice

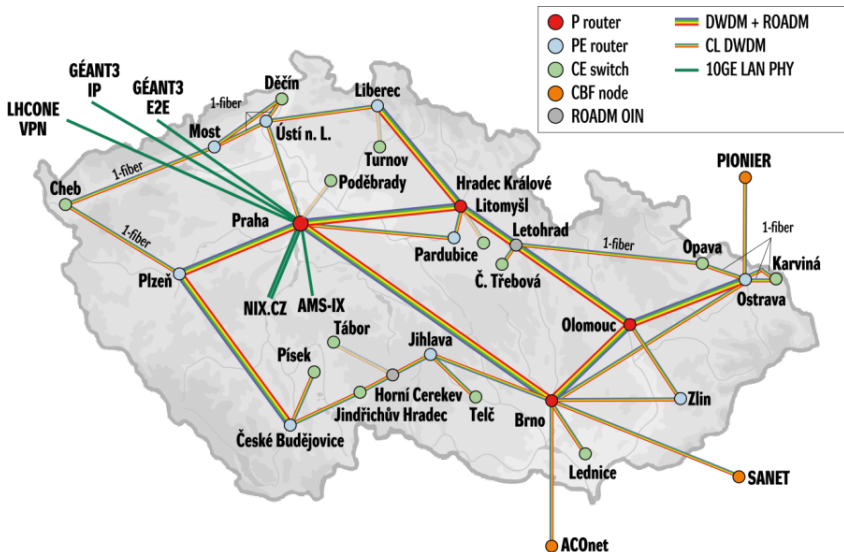
Kružnice (uzavřená cesta) v grafu je netriviální* neorientovaná cesta, která začíná i končí ve stejném vrcholu. *Orientovaná kružnice (také cyklus)* je kružnice složená z orientovaných hran respektující orientaci těchto hran.

* triviální cesta obsahuje jeden vrchol

Příklady

- V Internetu existuje mnoho redundantních linek – graf jeho fyzického propojení je cyklický.
- Lokální ethernetové sítě mají acyklickou topologii.
- Sítě založené na technologii Token Ring mají logickou kruhovou topologii.
- Sítě SONET podporují zapojení do kruhu.

Topologie sítě CESNET

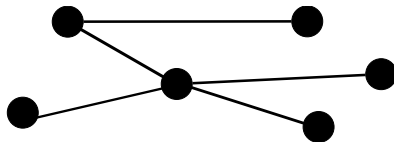


Definice

Les je graf, který neobsahuje kružnice. *Strom* je graf, který neobsahuje kružnice a je souvislý.

- Strom je tedy souvislý les.
- Lokální ethernetová síť je strom, protože je souvislá a acyklická.

Příklad stromu



Definice

*Strom, jehož hrany jsou orientované, se nazývá také **orientovaný**.*

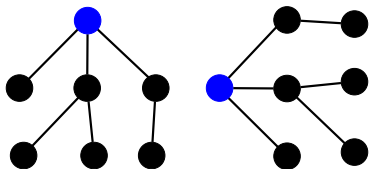
*Orientovaný strom, jenž má určen význačný vrchol (**kořen**) r , a v němž existuje orientovaná cesta z r do všech ostatních vrcholů, se nazývá **kořenový strom**.*

- Při kreslení kořenových grafů se obvykle vynechávají šipky definující orientaci hran, předpokládá se, že směřují „od kořene“.
- Vzhledem k absenci cyklů je interpretace jednoznačná.

Kde v sítích najdeme kořenové stromy?

- DNS – hierarchická struktura serverů obsluhujících domény různých úrovní.
- Multicast – zdroj je kořenem, cesty k příjemcům tvoří strom.

Dvě obvyklá kreslení kořenového stromu. Kořen je vyznačen modře.



Vztah orientovaných a kořenových stromů

Každý kořenový strom je orientovaný. Jaké jsou podmínky pro to, aby byl orientovaný strom zároveň kořenovým?

Věta

Orientovaný strom je kořenový právě tehdy, když právě jeden z jeho vrcholů má vstupní stupeň 0 a všechny ostatní vrcholy 1.

Důkaz.

⇒ Necht' r je kořen stromu a jeho vstupní stupeň je vyšší než 0. Potom do něj vede hrana z některého z ostatních vrcholů stromu. Do toho ovšem vede cesta z kořene, v grafu je tedy cyklus, čímž docházíme ke sporu.

Pokud do některého z ostatních vrcholů (označme jej u) vedou více než 2 hrany (z různých vrcholů v, w), potom do něj vedou 2 cesty z kořene, a to skrze cesty do v, w . Tím opět docházíme ke sporu. □

Vztah orientovaných a kořenových stromů – pokračování důkazu

Orientovaný strom je kořenový právě tehdy, když právě jeden z jeho vrcholů má vstupní stupeň 0 a všechny ostatní vrcholy 1.

Důkaz.

⇐ Označme r vrchol, jehož vstupní stupeň je 0. Poté pro každý jiný vrchol w platí následující:

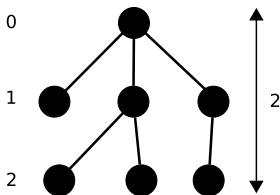
- Vstupní stupeň w je roven 1. Existuje tedy právě jeden vrchol, u_1 , z něž vede do w hrana. Není-li u_1 totožný s r , vede do něj opět hrana z právě jednoho vrcholu, u_2 . Takto tvořená řada vrcholů, z nichž vede cesta do w , je nutně konečná, neboť strom je acyklický, tudíž se v ní žádný z vrcholů nemůže opakovat. Posledním vrcholem v této posloupnosti musí být r .

Do každého vrcholu stromu tedy vede orientovaná cesta z vrcholu r a ten je tudíž kořenem stromu. □

Definice

Vzdálenost (počet hran na cestě) vrcholu od kořene stromu se nazývá **hloubka** či **úroveň** vrcholu.

- Hloubka kořene je rovna 0.
- Je zvykem kreslit vrcholy jedné úrovně ve stejné výšce.



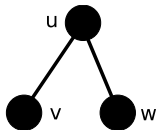
Definice

Výškou kořenového stromu označujeme nejvyšší z hloubek všech jeho vrcholů.

Definice

Vede-li v kořenovém stromu hrana z vrcholu u do v , nazývá se u **rodičem (otcem)** v a v **synem (potomkem)**.
Vrcholy mající společného rodiče nazýváme **sourozenci**.

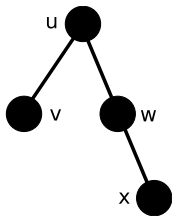
Vrchol u je rodičem obou vrcholů v, w , které jsou sourozenci.



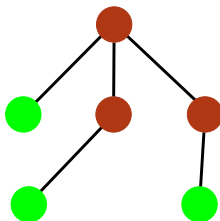
Definice

Vrchol u je **předkem** vrcholu v , pokud leží na cestě z r do v . v se v takovém případě nazývá **následníkem** vrcholu u .

Vrcholy v, w, x jsou následníky vrcholu u . Vrchol w má jediného následníka x . Předky x jsou u, w .



Listy jsou vyznačeny zeleně, vnitřní vrcholy stromu hnědě.



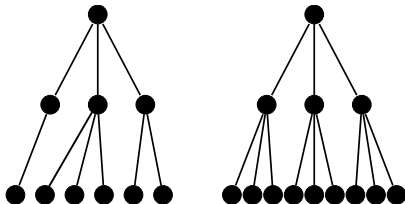
Definice

Vrchol, který nemá žádné potomky, nazýváme **list** stromu. Ostatní vrcholy se označují jako **vnitřní**.

Definice

Kořenový strom, jehož každý vrchol má nejvýše n potomků, se nazývá **n -ární**.
 n -ární strom, jehož vnitřní vrcholy mají právě n potomků a všechny listy jsou stejné hloubky, se nazývá **úplný n -ární**.

Levý strom je 3-ární (ternární), pravý je úplný ternární.



Uspořádané stromy

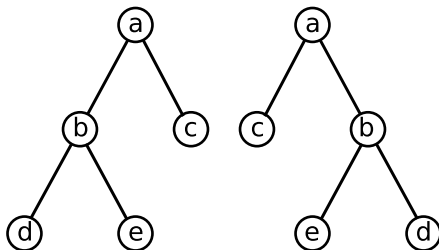
V některých případech může být výhodné potomky každého vrcholu jednoznačně pojmenovat a seřadit:

Definice

Uspořádaný strom je kořenový strom s daným pořadím potomků každého vrcholu.

Při kreslení uspořádaného grafu se dané pořadí vrcholů dodržuje ve směru zleva doprava.

Isomorfní * kořenové stromy s různým uspořádáním.



* Zopakujte si pojem isomorfismus.

Pro každou úroveň n -árního stromu je dán limit počtu vrcholů v této úrovni:

Věta

V m -té úrovni n -árního stromu se nachází nejvýše n^m vrcholů.

Důkaz.

Indukcí:

Pro kořen platí triviálně: $n^0 = 1$.

Nechť je v úrovni k právě n^k vrcholů. Každý z nich může mít nejvýše n potomků. Úroveň $k + 1$ tedy obsahuje nejvýše $n * n^k = n^{k+1}$ vrcholů. □

Speciálním (a prakticky nejpoužívanějším) n -árním stromem je strom **binární**.

Definice

*Uspořádaný 2-ární strom se nazývá **binární**. Potomci každého vrcholu jsou označováni jako **levý** a **pravý**.*

- Každý kořenový strom lze převést na binární.
- Vnitřní algoritmy směrovacích zařízení mohou být založeny na binárních stromech.

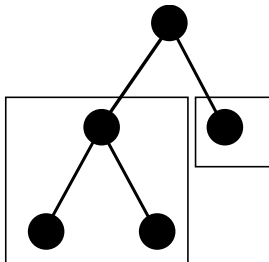
Definice

Indukovaný podgraf binárního stromu G , tvořený jedním potomkem vrcholu v a všemi jeho následníky, se nazývá **podstromem** vrcholu v a stromu G .*

- Podstrom binárního stromu je také binárním stromem.
- Každý vrchol má **levý** a **pravý** podstrom, přičemž jeho levý, resp. pravý, potomek je kořenem tohoto podstromu.
- Pravý a levý podstrom binárního stromu o výšce h mají výšku nejvýše $h - 1$, přičemž nejméně jeden z nich má výšku právě $h - 1$.

* Zopakujte si pojmy podgraf a indukovaný podgraf.

Obrázek : Levý a pravý podstrom kořene stromu.



Věta

Úplný binární strom výšky h má právě $2^{h+1} - 1$ vrcholů.

Důkaz.

Indukcí:

Pro binární strom výšky 0 platí zřejmě.

Nechť strom výšky k má právě $2^{k+1} - 1$ vrcholů. Jak bylo dokázáno dříve*,
($k + 1$)-ní vrstva obsahuje 2^{k+1} vrcholů. Strom výšky $k + 1$ tedy má

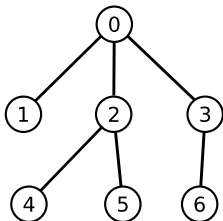
$$2^{k+1} - 1 + 2^{k+1} = 2 * 2^{k+1} - 1 = 2^{k+2} - 1$$

vrcholů. □

* Víme: V m -té úrovni n -árního stromu se nachází nejvýše n^m vrcholů.

Reprezentace kořenových stromů

- Kořenové stromy lze jednoznačně reprezentovat **polem rodičů** – tedy polem, ve kterém je pro každý vrchol uložen pouze název jeho rodiče.
- Taková reprezentace je prostorově velmi výhodná (lineární prostorová složitost).



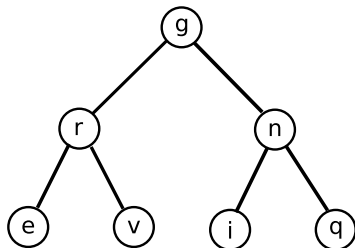
Pole rodičů má tvar: - 0 0 0 2 2 3.

Cvičení:

- Nakreslete kořenový strom podle zadaného pole rodičů:
- 0 1 1 2 2 3 4 4

Reprezentace úplných ohodnocených stromů

- Obdobně výhodně lze reprezentovat úplné ohodnocené stromy.
- Každý vrchol může mít v lineárním poli pevně danou pozici.
- Na této pozici je v poli uloženo ohodnocení vrcholu.
- Konkrétně pro binární strom:
 - Kořen je uložen na pozici 0.
 - Potomci vrcholu k jsou uloženi na pozicích $2 * k + 1, 2 * k + 2$.



Pole reprezentující tento binární graf obsahuje hodnoty $g \ r \ n \ e \ v \ i \ q$.

V některých případech (např. synchronizace distribuovaných algoritmů a výpočtů) je potřebné projít všemi vrcholy grafu a vykonat nějakou akci. Průchod binárním stromem je možné provést 2 základními způsoby:

- 1 průchod po úrovních
- 2 průchod po podstromech

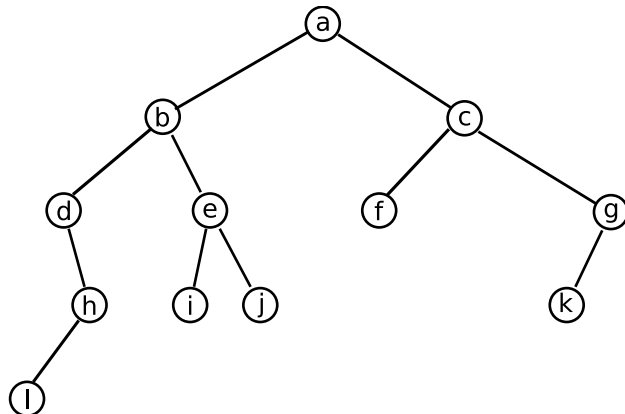
Průchod binárním stromem po úrovních

Vlož kořen do fronty.

Dokud je fronta neprázdná:

Odstraň její první vrchol a proved' akci.

Vlož do fronty jeho potomky v daném pořadí.



Vrcholy stromu jsou navštíveny v pořadí *a, b, c, d, e, f, g, h, i, j, k, l*.

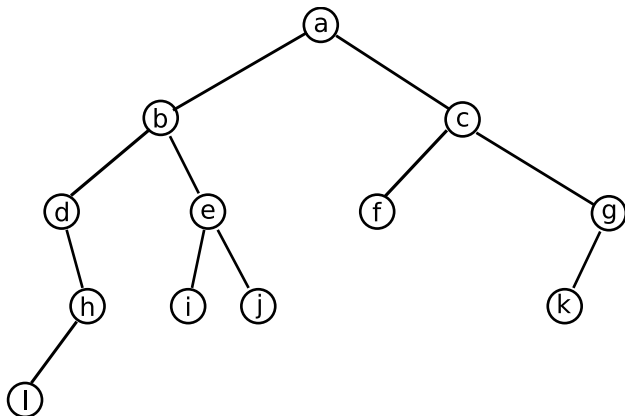
Proveď akci v kořenu stromu.

Spust' algoritmus průchodu v levém podstromu.

Spust' algoritmus průchodu v pravém podstromu.

- Tato verze algoritmu je rekurzivní. Průchod je možné implementovat iterativně bez rekurzivních volání za použití zásobníku.
- Akci lze také provádět po průchodu levým či oběma podstromy.

Průchod po podstromech – příklad



Vrcholy stromu jsou navštíveny v pořadí $a, b, d, h, l, e, i, j, c, f, g, k$.

Cvičení:

V jakém pořadí budou vypsány vrcholy, pokud bude výpis vrcholu proveden
1) po průchodu levým podstromem; 2) po průchodu oběma podstromy?