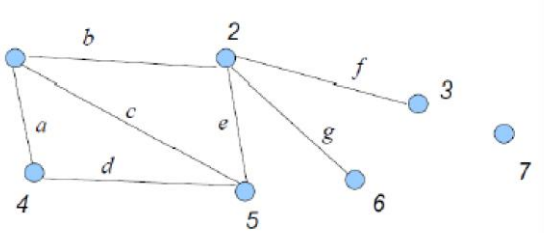
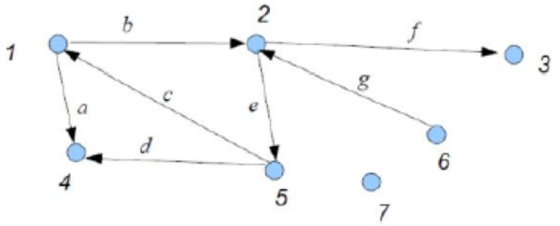
9. Obyčejné grafy (stupně uzlů, cesty a kružnice, souvislost grafu, stromy, kostry, Kruskalův a Primův algoritmus pro hledání minimální kostry ohodnoceného grafu).

Obyčejné grafy a jejich varianty

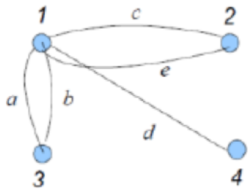
Obyčejný graf je dvojice $G = (U, H)$, kde U je konečná množina uzlů (vrcholů) a $H = \{\{u, v\} : u, v \in U \wedge u \neq v\}$ je konečná množina hran. O hraně $h=\{u,v\}$ říkáme, že je incidentní s uzly u a v, nebo že je mezi uzly u a v, spojuje u a v.



Orientovaný graf je dvojice $G = (U, H)$, kde U je konečná množina uzlů (vrcholů) a $H = \{\{u, v\} : u, v \in U \wedge u \neq v\}$ je konečná množina orientovaných hran.

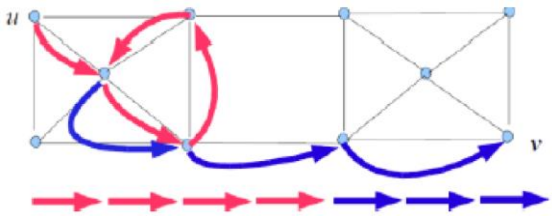


Obecný graf (multigraf) je trojice $G = (U, H, \varepsilon)$, kde U je konečná množina uzlů H je konečná množina hran a ε je zobrazení, které každé dvojici různých uzlů přiřazuje hranu ε: $\{\{u, v\} : u, v \in U \wedge u \neq v\} \rightarrow H$. Mezi jednou dvojicí uzlů tedy může být více hran.

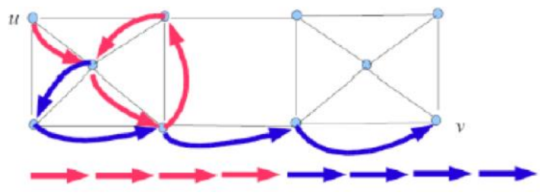


Průchod grafem

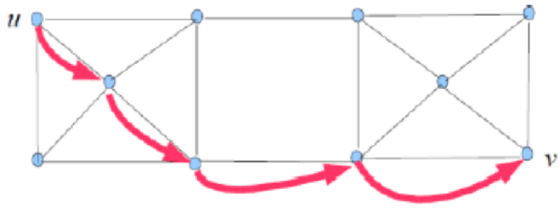
Je-li $G = (U, H)$ obyčejný graf, **sled** mezi uzly u a v o délce n je posloupnost $(u = w_0, h_1, w_1, h_2, \dots, w_{n-1}, h_n, w_n = v)$ takovou, že $w_0, w_1, \dots, w_n \in U$ a kde $h_1, w_2, \dots, h_n \in H$ a že každá hrana spojuje ve sledu dva sousední uzly $h_i = (w_{i-1}, w_i)$, $1 \leq i \leq n$. Opakovat se můžou uzly i hrany.



Je-li $G = (U, H)$ obyčejný graf, **tah** mezi uzly u a v o délce n je sled $(u = w_0, h_1, w_1, h_2, \dots, w_{n-1}, h_n, w_n = v)$ takový, že $\forall i, j \in <1, n> : i \neq j \Rightarrow h_i \neq h_j$. V tahu se tedy mohou opakovat uzly, ale už ne hrany.

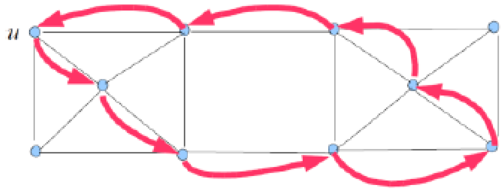


Je-li $G = (U, H)$ obyčejný graf, **cesta** mezi u a v o délce n je sled $(u = w_0, h_1, w_1, h_2, \dots, w_{n-1}, h_n, w_n = v)$ takový, že $\forall i, j \in <1, n> : i \neq j \Rightarrow w_i \neq w_j \wedge h_i \neq h_j$. V cestě se tedy nemohou opakovat ani uzly, ani hrany.

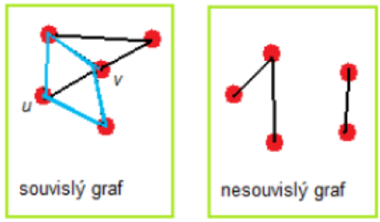


Je-li $G = (U, H)$ obyčejný graf, **kružnice** v grafu G o délce n je sled $(w_0, h_1, w_1, h_2, \dots, w_{n-1}, h_n, w_n)$

takový, že $\forall i, j \in <1, n-1> : i \neq j \Rightarrow w_i \neq w_j \wedge w_0 = w_n$. Kružnice má všechny hrany a uzly různé s výjimkou 1. a posledního.



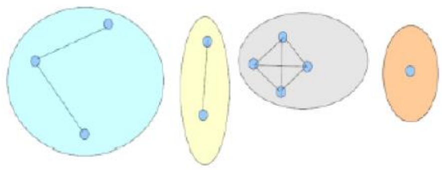
Části grafu



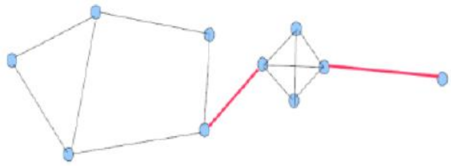
Je-li $G = (U, H)$ obyčejný graf, říkáme, že **je souvislý**, když pro $\forall u, v \in U$ existuje sled $(u = w_0, h_1, w_1, h_2, \dots, w_{n-1}, h_n, w_n = v)$.

Jsou-li $G = (U, H)$ a $G' = (U', H')$ obyčejnými grafy, pak říkáme, že G' **je podgrafem** G, když $U' \subseteq U \wedge H' \subseteq H$. Pokud navíc platí, $(u, v \in U' \wedge \{u, v\} \in H) \Rightarrow \{u, v\} \in H'$ nazývá se podgraf G' **faktorem**. Modrá část v souvislém grafu je podgrafem, není však faktorem, protože chybí hrana spojující u a v.

Jsou-li $G = (U, H)$ a $G' = (U', H')$ obyčejnými grafy, pak říkáme, že G' **je komponentou** G, když G' je souvislým faktorem grafu G a platí: $(U' \subset U'' \wedge G'' = (U'', H'')$ je podgraf $G) \Rightarrow G''$ není souvislý. Komponenta je tedy uzlově maximální souvislý faktor grafu.



Je-li $G = (U, H)$ obyčejný graf a $h \in H$, pak řekneme, že hrana h je **mostem**, pokud by se jejím odstraněním z grafu zvýšil počet komponent grafu. Pokud je hrana $h = \{u, v\}$ mostem, tak jejím odstraněním pak uzly u a v leží v různých komponentách.



Stupeň uzlu

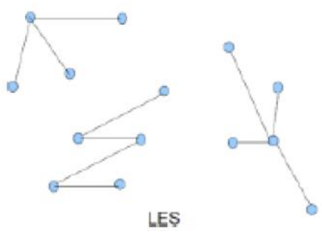
Je-li $G = (U, H)$ obyčejný graf a $u \in U$, pak definujeme číslo $\deg(u)$ jako **stupeň uzlu**, které nám říká počet hran incidentních s uzlem u.

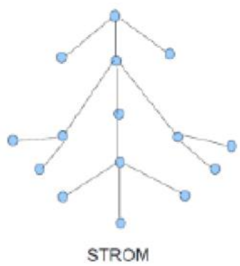
Nechť je $G = (U, H)$, kde $|H| = m$, pak vztah mezi sumou stupňů všech uzlů a počtem všech hran: $\Sigma_{u \in U} \deg(u) = 2m$.

Stromy

Obyčejný graf, jehož žádný podgraf není kružnicí, se nazývá **les**.

Obyčejný souvislý graf, jehož žádný podgraf není kružnicí, se nazývá **strom**.



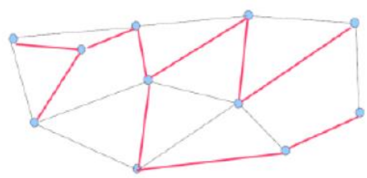


Nechť je $G = (U, H)$ je les, který má aspoň jednu hranu. Pak existují dva uzly $u, v \in U$ takové, že $\deg(u) = \deg(v) = 1$.

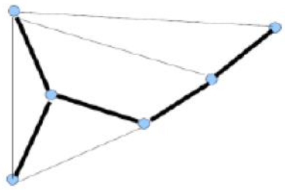
Nechť je $G = (U, H)$ je obyčejný graf a $|U| = n$, $|H| = m$. Pak jsou následující podmínky ekvivalentní G je strom $\Leftrightarrow$ G je souvislý a $m = n - 1$ $\Leftrightarrow$ G neobsahuje jako podgraf kružnici $\Leftrightarrow$ G je souvislý a každá hrana je mostem $\Leftrightarrow$ mezi každou dvojicí různých uzlů v G existuje jediná cesta.

Kostry

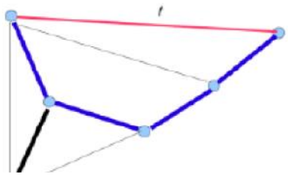
Nechť je $G = (U, H)$ je obyčejný graf, pak jeho podgraf $K = (U, H')$ nazveme **kostrou grafu** G, pokud je K stromem. Každá kostra grafu je tedy uzlově maximální strom obsažený jako podgraf v grafu G.



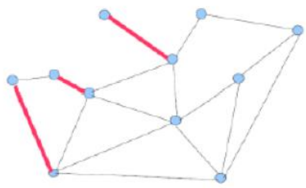
Nechť G je obyčejný graf, pak G je souvislý, právě když má kostru.



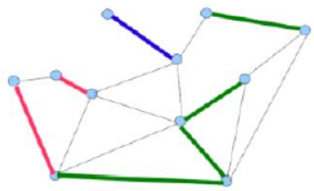
Nechť je $G = (U, H)$ je obyčejný graf a $K = (U, H')$ je jeho kostra. Potom $H' \subseteq H$ a hrany z H, které nejsou v H' se nazývají **tětivy kostry** K.



Nechť je $G = (U, H)$ je obyčejný graf a $K = (U, H')$ je jeho kostra a $t = \{u, v\}$ je tětiva kostry K. Pak podle závěrečných ekvivalentních tvrzení, platících pro stromy, existuje jediná cesta mezi uzly u a v v K. Tato cesta pak spolu s tětivou tvoří kružnici v grafu G, kterou nazýváme **základní kružnice kostry** K **vytvořená tětivou** t, což značíme $C^K(t)$.

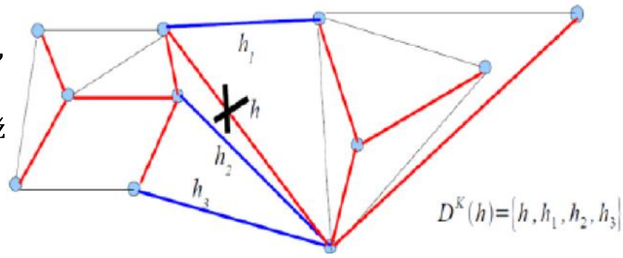


Nechť je $G = (U, H)$ je obyčejný souvislý graf a $D \subseteq H$ je podmnožina hran. Potom tuto podmnožinu nazveme **rozpojovací množinou grafu** G, pokud odebráním všech hran množiny D z H vznikne nesouvislý graf. Formálně je $G' = (U, H \setminus D)$ nesouvislý graf.



Nechť je $G = (U, H)$ je obyčejný souvislý graf a $D \subseteq H$ je podmnožina hran. Potom tuto množinu nazveme **řezem grafu** G, pokud D je množinově minimální rozpojující množinou grafu G, tedy pokud žádná její vlastní podmnožina není rozpojující množinou. Pokud D je řezem, pak odstraněním se graf rozpadá na komponenty.

Nechť je $G = (U, H)$ je obyčejný graf a $K = (U, H')$ je jeho kostra. Nechť odstraněním hrany h z jeho kostry K, vzniknou dva podgrafy $K_1 = (U_1, H_1)$ a $K_2 = (U_2, H_2)$ a nechť $D = \{h = \{u, v\} \mid h \in H \wedge u \in U_1 \wedge v \in U_2\}$ je takto vzniklý řez. Potom D nazýváme **základní řez kostry** K **vytvořený hranou** h, což značíme $D(h)$.



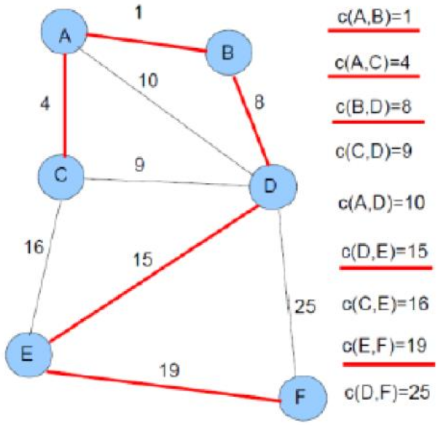
Ohodnocený graf

Nechť je $G = (U, H)$ je obyčejný graf. Je-li navíc dáno zobrazení $c\colon H \to \mathbb{R}$, potom trojici $G = (U, H, c)$ nazýváme **oceněným grafem**. Každé hraně h grafu G je tak přiřazeno reálné číslo $c(h)$, které nazýváme **cenou hrany** h. Je-li $G' = (U', H')$ podgraf grafu G, potom $c'\ G' = \Sigma_{h \in H'}\ c(h)$ nazýváme **cenou podgrafu** G'.

Nechť $G=(U,H,c)$ je obyčejný oceněný graf a $K=(U,H')$ je kostrou tohoto grafu. Pak říkáme, že K je **minimální kostrou grafu** G, jestliže platí $c(K) \leq c(L)$, kde L je libovolná kostra grafu G.

Algoritmy nalezení minimální kostry

Kruskalův algoritmus



Je dán oceněný obyčejný souvislý graf $G=(U,H,c)$, kde $|U|=n$ a $H=\{h1,h2,\ldots,hk\}$. Setřídíme hrany z H do posloupnosti $S=(s1,s2,\ldots sk)$,kde $c(si)\leq c(sj)$ pro $i<j$. Odstraníme první hranu $s1$ a vložíme ji do vznikající kostry grafu K. Takto pokračujeme v odstraňování hran z S a vkládáme je do K jen v případě, že by nevzniknula v K kružnice, jinak takovou hranu přeskočíme. Algoritmus ukončíme ve chvíli, kdy je K kostrou grafu (obsahuje všechny uzly z U).

Primův algoritmus

Je dán obyčejný oceněný souvislý graf $G=(U,H,c)$. Pro podgraf $K=(V,J)$ grafu G, který neobsahuje kružnici, označme $K^+=(V^+,J^+)$ graf, který vznikne z K přidáním uzlu $u\in V$ do V^+ a hrany $h\in J$ do J^+ takové, že:

- h je incidentní s u a s nějakým jiným uzlem z V;
- h nevytvoří v K^+ kružnici;
- h je nejmenší hranou s výše jmenovanými dvěma vlastnostmi.

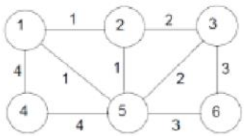
Primův algoritmus vyjde z libovolného bodu a postupně se přidává vždy hrana s nejmenší cenou taková, že předchozí graf rozšíří tak, aby byl souvislý a přitom neobsahoval kružnici. Oproti Kruskalovu algoritmu má tu výhodu, že se nemusejí předem seřazovat podle vzrůstající ceny všechny hrany. Při Kruskalově algoritmu se totiž většinou hrany s vysokými cenami vůbec nevyužijí.

Pokud jsou ceny hran grafu $G=(U,H,c)$ kde $U=\{u1,u2,\ldots,un\}$ zadány ve formě matice, kde prvek na i-tém řádku a v j-tém sloupci označuje hrany incidentní s uzly u_i a u_j. Pak je možno Primův algoritmus vyjádřit v následující formě:

(1) Vyškrtnou se všechny prvky v prvním sloupci a označíme první řádek;

(2) Pokud v označených řádcích neexistuje žádný nepodtržený prvek, algoritmus končí a podtržené prvky označují hrany v minimální kostře. Jinak se vybere minimální prvek;

(3) Je-li vybraný prvek cij, podtrhne se, označí se i-tý řádek a vymažou se nepodtržené prvky j-tého sloupce. Vrátíme se ke kroku (2).



$$C_1=\left|\begin{matrix}- & 1 & - & 4 & 1 & -\\ - & - & 2 & - & 1 & -\\ - & 2 & - & - & 2 & 3\\ - & - & - & - & 4 & -\\ - & 1 & 2 & 4 & - & 3\\ - & - & 3 & - & 3 & -\end{matrix}\right|\begin{matrix}\cdot\\ \\ \\ \\ \\ \\ \end{matrix}\quad C_2=\left|\begin{matrix}- & \underline{1} & - & 4 & 1 & -\\ - & - & 2 & - & 1 & -\\ - & - & - & - & 2 & 3\\ - & - & - & - & 4 & -\\ - & - & 2 & 4 & - & 3\\ - & - & 3 & - & 3 & -\end{matrix}\right|\begin{matrix}\cdot\\ \cdot\\ \\ \\ \\ \\ \end{matrix}\quad C_3=\left|\begin{matrix}- & \underline{1} & - & 4 & \underline{1} & -\\ - & - & 2 & - & - & -\\ - & - & - & - & - & 3\\ - & - & - & - & - & -\\ - & - & - & 4 & - & 3\\ - & - & 3 & - & - & -\end{matrix}\right|\begin{matrix}\cdot\\ \cdot\\ \\ \\ \cdot\\ \\ \end{matrix}$$

$$C_4=\left|\begin{matrix}- & \underline{1} & - & 4 & \underline{1} & -\\ - & - & \underline{2} & - & - & -\\ - & - & - & - & - & 3\\ - & - & - & - & - & -\\ - & - & - & 4 & - & 3\\ - & - & - & - & - & -\end{matrix}\right|\begin{matrix}\cdot\\ \cdot\\ \cdot\\ \\ \cdot\\ \\ \end{matrix}\quad C_5=\left|\begin{matrix}- & \underline{1} & - & 4 & \underline{1} & -\\ - & - & \underline{2} & - & - & -\\ - & - & - & - & - & \underline{3}\\ - & - & - & - & - & -\\ - & - & - & 4 & - & -\\ - & - & - & - & - & -\end{matrix}\right|\begin{matrix}\cdot\\ \cdot\\ \cdot\\ \\ \cdot\\ \cdot\end{matrix}\quad C_6=\left|\begin{matrix}- & \underline{1} & - & \underline{4} & \underline{1} & -\\ - & - & \underline{2} & - & - & -\\ - & - & - & - & - & \underline{3}\\ - & - & - & - & - & -\\ - & - & - & - & - & -\\ - & - & - & - & - & -\end{matrix}\right|\begin{matrix}\cdot\\ \cdot\\ \cdot\\ \\ \cdot\\ \cdot\end{matrix}$$