2.3 Nejkratší cesty.

Je dán prostý orientovaný graf $G = (V, E)$, kde $V = \{1, 2, \dots, n\}$, a ohodnocení hran a, tj. zobrazení $a\colon E \to \mathbb{Z}$.

2.3.1 Matice délek A. *Matice délek* je čtvercová matice $\mathbf{A} = (a(i,j))$ řádu n, kde n je počet vrcholů grafu G, a

$$a(i,j) = \begin{cases} 0, & \text{pro } i = j \\ a(e), & \text{pro } e = (i,j) \in E \\ \infty, & \text{pro } (i,j) \notin E \end{cases}$$

2.3.2 Matice vzdáleností U. *Matice vzdáleností* je čtvercová matice $\mathbf{U} = (u(i,j))$ řádu n, kde n je počet vrcholů grafu G, a

$$u(i,j) = \begin{cases} 0, & \text{pro } i = j, \\ \text{délka nejkratší cesty z } i \text{ do } j, & \text{jestliže existuje cesta z } i \text{ do } j \\ \infty, & \text{jestliže neexistuje cesta z } i \text{ do } j \end{cases}$$

2.3.3 Pozorování. Předpokládejme, že vrchol y je orientovaně dostupný z vrcholu x v grafu G. Pak platí:

1. Jestliže graf G obsahuje pouze cykly kladné délky (tj. neobsahuje ani cykly záporné délky ani nulové délky), pak nejkratší sled z vrcholu x do vrcholu y existuje a je současně nejkratší cestou z x do y.
2. Jestliže v grafu G neexistuje cyklus záporné délky, pak nejkratší sled z x do y má stejnou délku jako nejkratší cesta z x do y.
3. Jestliže v grafu G neexistuje cyklus záporné délky, pak pro každý sled C z x do y existuje cesta z x do y, která je kratší nebo stejně dlouhá jako sled C.

2.3.4 Trojúhelníková nerovnost. Jestliže v grafu G neexistuje cyklus záporné délky, pak pro každé tři vrcholy x, y, z platí

$$u(x,y) \le u(x,z) + u(z,y).$$

Důkaz: Jestliže vrchol z není orientovaně dostupný z vrcholu x nebo vrchol y není orientovaně dostupný z vrcholu z, pak trojúhelníková nerovnost triviálně platí.

V opačném případě označme P_1 nejkratší cestu z vrcholu x do vrcholu z a P_2 nejkratší cestu z vrcholu z do vrcholu y. Spojení obou cest je sled P_1, P_2 s délkou rovnou součtu délek cest P_1 a P_2. Protože graf neobsahuje cykly záporné délky, tento sled obsahuje cestu, která je kratší nebo stejně dlouhá jako délka P, tj. jako $u(x,y) + u(z,y)$. Proto i pro délku nejkratší cesty z x do y platí $u(x,y) \le u(x,z) + u(z,y)$.

2.3.5 Bellmanův princip optimality. Jestliže v grafu G neexistuje cyklus záporné délky, pak pro každé tři vrcholy x, y, z platí

$$u(x,y) = \min_{z \neq y}(u(x,z) + a(z,y)).$$

Důkaz: Vztah jistě platí pro vrcholy x, y, pro které neexistuje cesta z x do y.

Předpokládejme, že existuje cesta z x do y, tj. $u(x,y) < \infty$. Protože $u(z,y) \leq a(z,y)$ pro každé dva vrcholy z, y, víme z trojúhelníkové nerovnosti, že $u(x,y) \leq u(x,z) + a(z,y)$. Proto

$$u(x,y) \leq \min_{z \neq y}(u(x,z) + a(z,y)).$$

Rovnost nastává pro vrchol z, který je předposlední na nejkratší cestě z vrcholu x do vrcholu y.

2.3.6 Nejkratší cesty z výchozího vrcholu r. Úloha: Najděte délky nejkratších cest z výchozího vrcholu r.

2.3.7 Obecné schema.

Vstup: orientovaný graf $G = (V, E)$ a ohodnocení hran a.

Výstup: hodnoty $U(v)$ rovné $u(r, v)$.

1. (Inicializace.)
   $U(r) := 0$, $U(v) := \infty$ pro $v \neq r$;
2. (Zpracování hran.)
   Existuje-li hrana $e = (v, w)$ taková, že
   $U(w) > U(v) + a(e)$
   položíme $U(w) := U(v) + a(e)$.
3. (Ukončení.)
   Jestliže $U(w) \leq U(v) + a(e)$ pro každou hranu $e = (v, w)$, stop;
   jinak pokračujeme krokem 2.

2.3.8 Tvrzení. Jestliže v grafu G neexistuje cyklus záporné délky a hodnota $U(v) \neq \infty$, pak $U(v)$ je délka některé cesty z vrcholu r do vrcholu v.

Nástin důkazu: Označme $U_t(y)$ hodnotu $U(y)$ v okamžiku t. Platí: jestliže v nějakém okamžiku t_k je $U_{t_k}(x) < \infty$, tak musí existovat sled z r do x

$$r = v_1, e_1, v_2, e_2, \ldots, v_{k-1}, e_{k-1}, v_k = x$$

a časové okamžiky $t_1 < t_2 < \ldots < t_k$ tak, že

$$U_{t_i}(v_i) = \sum_{j=1}^{i} a(e_j).$$

Nyní je třeba dokázat, že se nejedná o sled, ale o cestu. Kdyby se ve sledu opakoval vrchol, tj. kdyby např. $v_i = v_j$ pro $i < j$, pak $U_{t_i}(v_i) > U_{t_j}(v_j)$ a proto se dá dokázat, že $v_i, e_i, v_{i+1}, e_{i+1}, \ldots, v_j$ obsahuje cyklus záporné délky.

2.3.9 Věta. Jestliže graf G neobsahuje cyklus záporné délky a hodnoty $U(v)$ byly získány podle schematu 2.3.7, pak $U(v) = u(r, v)$.

Důkaz: Sporem. Kdyby tvrzení věty neplatilo, po skončení práce schematu by existoval vrchol v takový, že $U(v) > u(r, v)$. To také znamená, že $u(r, v) < \infty$. Vezměme nejkratší cestu P z vrcholu r do vrcholu v. Protože $U(r) = u(r, r)$ a poslední vrchol je v, pro který $U(v) > u(r, v)$, na cestě P existuje hrana $e = (x, y)$ taková, že $U(x) = u(r, x)$ a $U(y) > u(r, y)$. Vezměme první takovou hranu $e = (x, y)$. Pro tyto vrcholy x, y platí:

$$U(y) > u(r, y) = u(r, x) + a(x, y) = U(x) + a(x, y).$$

Tedy, obecné schema nemělo skončit, protože trojúhelníková nerovnost neplatí pro hranu $e = (x, y)$. Tedy není pravda, že se postup zastavil.

2.3.10 Nejprve uvedeme jednoduchý algoritmus, nazveme ho Algoritmus I, který vychází z obecného schematu 2.3.7. Vždy probereme všechny hrany grafu v libovolném, ale pevně daném pořadí. Jestliže při průchodu nedojde ke změně žádné hodnoty $U(x)$, pak už platí trojúhelníková nerovnost pro všechny hrany a můžeme algoritmus ukončit.

Protože cesta v grafu s n vrcholy má nejvýše $n-1$ hran, nemá-li graf záporné cykly, musí následující algoritmus skončit po nejvýše n průchodech krokem 2. Fakt, že průchodů krokem 2 je maximálně n dává invariant tohoto algoritmu; je to $n - k$ kde k je počet již proběhlých kroků 2.

Dále nám toto pozorování umožňuje poznat graf se zápornými cykly. Jestliže i při n-tém průchodu krokem 2 došlo ke změně některé hodnoty $U(x)$, pak graf obsahuje cyklus záporné délky a výsledky, které jsme algoritmem dostaly, jsou nesprávné.

Poznamenejme, že časové nároky algoritmu I jsou $\mathcal{O}(m.n)$, kde $n = |V|$ a $m = |E|$.

2.3.11 Algoritmus I.

Vstup: orientovaný graf $G = (V, E)$ a ohodnocení hran a.

Výstup: hodnoty $U(v)$ rovné $u(r, v)$.

1. (Inicializace.)
   $U(r) := 0$, $U(v) := \infty$ pro $v \neq r$;
2. (Zpracování hran.)
   Pro každou hranu $e \in E$ provedeme
   jestliže $U(KV(e)) > U(PV(e)) + a(e)$
   položíme $U(KV(e)) := U(PV(e)) + a(e)$.
3. (Ukončení.)
   Jestliže během kroku 2 nedošlo ke změně hodnoty $U(v)$, stop, a vrátíme $U(v)$.
   Jinak pokračuj krokem 2.

2.3.12 Při práci algoritmu I se může stát, že při nevhodné volbě pořadí hran první průchod krokem 2 změní jen málo třeba i jen jednu hodnotu $U(x)$ — to nastane v případě, že z vrcholu r vychází jen jedna hrana a ta bude probírána jako poslední. Uvedeme proto ještě sofistikovanější variantu schematu 2.3.7 — jedná se o algoritmus II.

V tomto algoritmu udržujeme množinu M vrcholů „podezřelých“ z toho, že pro hrany, které z nich vycházejí by nemusela platit trojúhelníková nerovnost. Jinými slovy, jestliže $x \notin M$, pak pro každou hranu e s $PV(e) = x$ již trojúhelníková nerovnost platí.

Na začátku práce je $M = \{r\}$. Množinu M udržujeme tak, že kdykoli snižujeme hodnotu $U(x)$ pro nějaký vrchol x, vrchol x do množiny M zařadíme.

2.3.13 Algoritmus II.

Vstup: orientovaný graf $G = (V, E)$ a ohodnocení hran a.

Výstup: hodnoty $U(v)$ rovné $u(r, v)$.

1. (Inicializace.)
   $U(r) := 0$, $U(v) := \infty$ pro $v \neq r$; $M := \{r\}$
2. (Zpracování hran.)
   Dokud $M \neq \emptyset$, vybereme $x \in M$;
      $M := M \setminus \{x\}$
      pro každou hranu e s $PV(e) = x$ provedeme
         jestliže $U(KV(e)) > U(x) + a(e)$
         položíme $U(KV(e)) := U(x) + a(e)$; $M := M \cup \{KV(e)\}$.
3. (Ukončení.)
   Vrátíme $U(v)$; stop.

2.3.14 Nejkratší cesty mezi všemi dvojicemi vrcholů. Úkolem je najít celou matici vzdáleností (a ne jen jeden její řádek).

Množinu vrcholů grafu G označíme $V = \{1, 2, \dots, n\}$. Floydův algoritmus (v literatuře též nazývaný Floyd-Warshallův algoritmus) je založen na konstrukci matic $\mathbf{U}_k = (u_k(i, j))$ řádu n pro $k = 0, 1, \dots, n$ s následující vlastnosti:

$u_k(i, j)$ je délka nejkratší cesty z i do j, která prochází pouze vrcholy $1, 2, \dots, k$.

2.3.15 Tvrzení. Platí

1. $\mathbf{U}_0$ je matice délek $\mathbf{A}$.
2. $\mathbf{U}_n$ je matice vzdáleností $\mathbf{U}$.
3. Matici $\mathbf{U}_{k+1}$ získáme z matice $\mathbf{U}_k$ takto:

$$u_{k+1}(i, j) = \min\{u_k(i, j), u_k(i, k + 1) + u_k(k + 1, j)\}.$$

Důkaz: První dvě vlastnosti jednoduše vyplývají z definice matic $\mathbf{U}_0$ a $\mathbf{U}_n$.

Třetí vlastnost dostaneme, když si uvědomíme, že nejkratší cesta z i do j, která vede pouze přes vrcholy $1, 2, \dots, k + 1$ se buď vrcholu $k + 1$ vyhne (a pak je délky $u_k(i, j)$), nebo vrcholem $k + 1$ prochází a pak je délky $u_k(i, k + 1) + u_k(k + 1, j)$.

2.3.16 Floydův algoritmus.

Vstup: matice délek $\mathbf{A}$.

Výstup: matice vzdáleností $\mathbf{M} = \mathbf{U}$.

1. *[Inicializace]*
   $\mathbf{M} := \mathbf{A}$
2. `begin`
   `for` $k = 1, 2, \ldots, n$ `do`
   `for` $i = 1, 2, \ldots, n$ `do`
   `for` $j = 1, 2, \ldots, n$ `do`
   `begin`
   `if` $M(i,j) > M(i,k) + M(k,j)$ `then`
   $M(i,j) := M(i,k) + M(k,j)$
   `end`
   `end`

2.3.17 Ukončení Floydova algoritmu je zaručeno tím, že vnější cyklus se provádí n-krát, tj. variant je k, které se roste od 1 do n.

Invariantem je 2.3.14 a vlastnost 3 z 2.3.15.