Cvičení týden 10
Nejkratší cesty

Obsahem cvičení

bude řešení kontrolních úloh

z příprav na přednášky číslo 7 a 8

Používané pojmy:

- nejkratší cesty 1 $\rightarrow$ n, relaxace :
  - Dijkstrův algoritmus, Bellmanův-Fordův algoritmus
- nejkratší cesty n $\rightarrow$ n :
  - Floydův-Warshallův algoritmus, Johnsonův algoritmus, zobecněný F-W algoritmus

Kontrolní otázky

7.1 Která část Dijkstrova algoritmu je podstatně závislá na předpokladu nezáporného ohodnocení hran? Ukažte na jednom příkladu, že pro záporně ohodnocené hrany může Dijkstrův algoritmus dát špatný výsledek, a na jiném příkladu, že může dát správný výsledek.

7.2 Je možné prohlásit, že Dijkstrův algoritmus bude fungovat správně i při záporném ohodnocení hran, pokud bude zadaný graf acyklický?

7.3 Je možné prohlásit, že Dijkstrův algoritmus bude fungovat správně i při záporném ohodnocení hran, pokud bude použit k určení vzdáleností z kořene do ostatních uzlů kořenového stromu?

7.4 Navrhněte časově efektivní algoritmus pro určení celkového počtu různých orientovaných cest v acyklickém grafu.

(Návod: Inspirujte se algoritmem DAG-Paths a za hodnotu d[u] berte počet cest končících v uzlu u.)

7.5 Navrhněte algoritmus, který určí vzdálenost ze všech uzlů do uzlu s v acyklickém orientovaném grafu G. Určete potřebné datové struktury a časovou složitost navrženého algoritmu.

7.6 Navrhněte algoritmus lineární složitosti pro hledání nejdelších cest z daného uzlu do všech ostatních uzlů v acyklickém grafu.

Kontrolní otázky

7.7 Doplňte Dijkstrův a Bellman-Fordův algoritmus o výpočet hodnoty r[u], která představuje počet hran nejkratší cesty z uzlu s do uzlu u.

(Návod: Stačí vhodně upravit operace InitPaths a Relax.)

Kontrolní otázky

8.1 Podobně jako je při provádění Floyd-Warshallova algoritmu možné počítat matici P předchůdců uzlů na nejkratších cestách, je možné také počítat matici Q následníků uzlů na nejkratších cestách. Určete pravidlo, podle něhož se nastaví počáteční hodnoty prvků $q_{ij}^{(0)}$ této matice, a pravidlo pro přechod od (k-1)-ní ke k-té iteraci hodnot q_{ij}.

Algoritmus Floyda-Warshalla

```
D(0) = W;
for (k=1; k<=n; k++) {
  for (i=1; i<=n; i++)   {
     for (j=1; j<=n; j++) {
          dij(k) = min (dij(k-1) , dik(k-1) + dkj(k-1));
} } }
return D(n);
```

Výpočet matice předchůdců:

$p_{ij}^{(0)}$ = **0/null** pro $i=j$ nebo $w_{ij} = \infty$

= **i** jinak (pro $(u_i,u_j) \in H$)

$p_{ij}^{(k)}$ = $p_{ij}^{(k-1)}$ pro $d_{ij}^{(k-1)} \le d_{ik}^{(k-1)} + d_{kj}^{(k-1)}$

= $p_{kj}^{(k-1)}$ pro $d_{ij}^{(k-1)} > d_{ik}^{(k-1)} + d_{kj}^{(k-1)}$

Kontrolní otázky

8.2 Dalším rozšířením Floyd-Warshallova algoritmu zajistěte, aby po ukončení výpočtu byl znám počet hran na nejkratších cestách mezi všemi uzly (opět ve formě matice označené např. R). (Návod: Inspirujte se řešením obdobného problému pro algoritmus Dikstrův a Bellman-Fordův.)

Připomínka:

5 $d_{ij}^{(k)} = \min (d_{ij}^{(k-1)}, d_{ik}^{(k-1)} + d_{kj}^{(k-1)});$

Výpočet matice předchůdců:

$p_{ij}^{(0)} = $ **0/null** pro $i=j$ nebo $w_{ij} = +\infty$

i jinak (pro $(u_i,u_j) \in H$)

$p_{ij}^{(k)} = p_{ij}^{(k-1)}$ pro $d_{ij}^{(k-1)} \leq d_{ik}^{(k-1)} + d_{kj}^{(k-1)}$

$p_{kj}^{(k-1)}$ pro $d_{ij}^{(k-1)} > d_{ik}^{(k-1)} + d_{kj}^{(k-1)}$

Kontrolní otázky

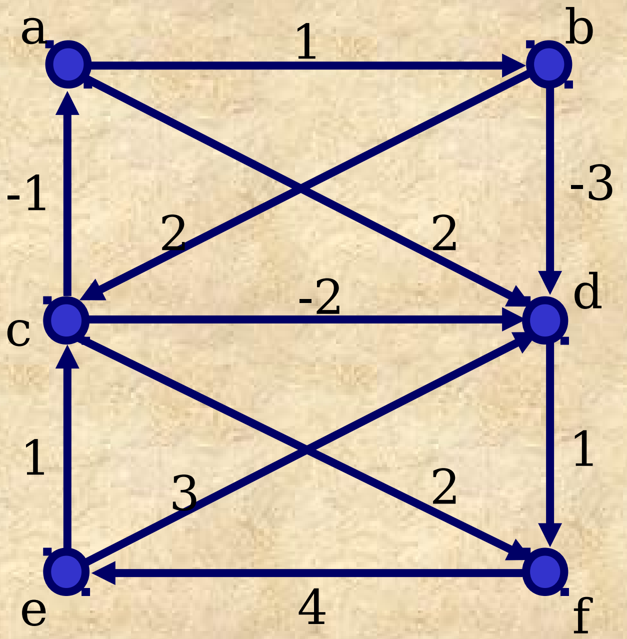
8.3 Zdůvodněte, proč je provádění Floyd-Warshallova algoritmu možné všechny iterace matice $D^{(k)}$ uchovávat v jediném poli.

(Návod: Ověřte, že vnitřní dva cykly nemění hodnotu prvků v k-tém řádku a k-tém sloupci, na nichž závisí hodnoty prvků v nové iteraci.)

8.4 Jak se při použití Floyd-Warshallova algoritmu zjistí případná existence záporných cyklů v grafu?

Kontrolní otázky

8.5 Pomocí Johnsonova algoritmu určete matici vzdáleností pro následující orientovaný graf :



Kontrolní otázky

8.6 **Jaký vztah platí mezi ohodnoceními $w(u,v)$ a $w'(u,v)$, pokud jsou hodnoty $w(u,v) \geq 0$ pro všechny hrany (u,v) ?**

8.7 **Jak je třeba definovat operace $\oplus$ a $\otimes$ a nosič (tj. množinu P) v odpovídajícím polookruhu, aby zobecněný Floyd-Warshallův algoritmus určil počet různých spojení mezi jednotlivými dvojicemi uzlů?**

$\boldsymbol{P} = \langle P, \oplus, \otimes, 0, 1 \rangle$:

$a \oplus (b \oplus c) = (a \oplus b) \oplus c$	$\langle P, \oplus, 0 \rangle$
$a \oplus 0 = 0 \oplus a = a$	je komutativní
$a \oplus b = b \oplus a$	monoid
$a \oplus a = a$	idempotence

$a \otimes (b \otimes c) = (a \otimes b) \otimes c$	$\langle P, \otimes, 1 \rangle$
$a \otimes 1 = 1 \otimes a = a$	je monoid
$a \otimes 0 = 0 \otimes a = 0$	s nulou

$a \otimes (b \oplus c) = (a \otimes b) \oplus (a \otimes c)$	distributivnost	uzávěr
$(b \oplus c) \otimes a = (b \otimes a) \oplus (c \otimes a)$	zleva a zprava	$a^* = 1 \oplus a \oplus a \otimes a \oplus a \otimes a \otimes a \oplus \dots$