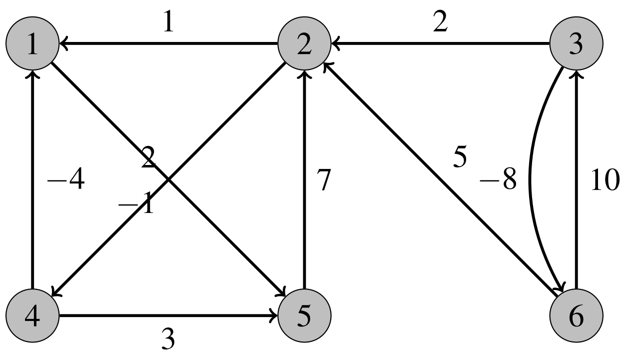
GRAFOVÉ ALGORITMY - CVIČENÍ 9: NEJKRATŠÍ CESTY ZE VŠECH DO VŠECH UZLŮ

Převzato z Cormen, Leiserson, Rivest, Stein: Introduction to algorithms. The MIT Press and McGraw-Hill, 2001; kapitola 25.

Příklad 1. Aplikujte algoritmus SLOW-ALL-PAIRS-SHORTEST-PATHS na ohodnocený orientovaný graf na obrázku 1. Demonstrujte mezivýsledné matice jednotlivých iterací algoritmu. Poté aplikujte a stejným způsobem demonstrujte algoritmus FASTER-ALL-PAIRS-SHORTEST-PATHS.



Obrázek 1: Příklad 1 a 11

Příklad 2. Zdůvodněte požadavek na matici sousednosti, která pro všechny $1 \leq i \leq n$ pokládá $w_{ii} = 0$. (n značí počet uzlů grafu)

Příklad 3. Uvažujme matici

$$L^{(0)} = \begin{pmatrix} 0 & \infty & \infty & \cdots & \infty \\ \infty & 0 & \infty & \cdots & \infty \\ \infty & \infty & 0 & \cdots & \infty \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \infty & \infty & \infty & \cdots & 0 \end{pmatrix}$$

Čemu odpovídá použití této matice z algoritmu SHORTEST-PATHS při klasickém násobení matic?

Příklad 4. Dokažte, že operace podobná násobení matic z algoritmu EXTEND-SHORTEST-PATHS je asociativní.

Příklad 5. Ukažte, jak vyjádřit problém nejkratší cesty z jednoho uzlu do všech uzlů jako násobení matic a vektoru. Popište, jak vyhodnocování tohoto násobení odpovídá (je podobné) algoritmu BELLMAN-FORD.

Příklad 6. Přepokládejme, že chceme, aby probrané algoritmy generovaly také seznamy uzlů na nejkratších cestách. Ukažte, jak s časovou složitostí $O(n^3)$ vypočítat matici předchůdců Π z výsledné matice L s váhami nejkratších cest.

Příklad 7. Jako rozšíření předchozího příkladu mohou být uzly na nejkratších cestách zpočteny už během výpočtu vah nejkratších cest. Definujme tedy $\pi_{ij}^{(m)}$ jako předka uzlu j na nějaké nejkratší cestě z i do j, která obsahuje nejvýše m hran. Modifikujte a zkombinujte algoritmy EXTEND-SHORTEST-PATHS a SLOW-ALL-PAIRS-SHORTEST-PATHS tak, aby počítaly matice $\Pi(1), \Pi(2), \ldots, \Pi(n-1)$ zároveň s maticemi $L(1), L(2), \ldots, L(n-1)$.

Příklad 8. Prostor potřebný pro algoritmus FASTER-ALL-PAIRS-SHORTEST-PATHS je $\lceil lg(n-1) \rceil$ matic, kdy každá matice obsahuje n^2 elementů, což dává celkovou prostorovou složitost $\Theta(n^2\, lg\, n)$. Modifikujte tuto proceduru tak, aby vyžadovala pouze $\Theta(n^2)$ prostoru, a to navíc použitím pouze dvou matic řádu n.

Příklad 9. Modifikujte algoritmus FASTER-ALL-PAIRS-SHORTEST-PATHS tak, aby dokázal detekovat přítomnost negativně ohodnoceného cyklu.

Příklad 10. Vytvořte algoritmus, který pro zadaný graf vrátí délku (počet hran) negativního cyklu s nejmenším počtem hran.

Floyd-Warshallův algoritmus

Příklad 11. Na ohodnocený orientovaný graf z obrázku 1 aplikujte Floyd-Warshallův algoritmus. Ukažte matice $D^{(k)}$ jako mezivýsledky jednotlivých iterací vnějšího cyklu algoritmu.

Příklad 12. Ukažte, jak vypočítat tranzitivní uzávěr grafu použitím technik z algoritmů EXTEND-SHORTEST-PATHS, SLOW-ALL-PAIRS-SHORTEST-PATHS a FASTER-ALL-PAIRS-SHORTEST-PATHS.

Příklad 13. Uvažujme změnu algoritmu FLOYD-WARSHALL vypuštěním všech horních indexů (0), (k), $(k-1)$ a (n). Zdůvodněte, proč tento modifikovaný algoritmus stále funguje, i když má prostorovou složitost $\Theta(n^2)$.

Příklad 14. Předpokládejme následující modifikaci definice $\pi_{ij}^{(k)}$:

$$\pi_{ij}^{(k)} = \begin{cases} \pi_{ij}^{(k-1)} & \text{if } d_{ij}^{(k-1)} < d_{ik}^{(k-1)} + d_{kj}^{(k-1)}, \\ \pi_{kj}^{(k-1)} & \text{if } d_{ij}^{(k-1)} \geq d_{ik}^{(k-1)} + d_{kj}^{(k-1)}. \end{cases}$$

Je tato alternativní definici z pohledu generování matice předchůdců Π korektní?

Příklad 15. Jak může být výstup Floyd-Warshallova algoritmu využit při detekci negativních cyklů v grafu?

Příklad 16. Navrhněte algoritmus pro výpočet tranzitivního uzávěru orientovaného grafu $G = (V, E)$ s časovou složitostí $O(V E)$.

Příklad 17. Předpokládejme, že pro orientovaný acyklický graf lze tranzitivní uzávěr vypočítat během $f(|V|,|E|)$ kroků, kde f je nějaká monotónně rostoucí funkce. Ukažte, že pak čas na výpočet tranzitivního uzávěru $G^* = (V,E^*)$ pro obecný orientovaný graf $G = (V,E)$ je $f(|V|,|E|) + O(V + E^*)$.