

EADŠ - cvičenie 12

8. decembra 2022

Grafy - opakovanie

Čo je graf?

- ▶ Intuitívne: "krúžky a čiariočky"
- ▶ Formálne: $G = (V, U)$ (množina vrcholov a hrán)

Čo je hrana?

- ▶ Intuitívne: "čiaročka z jedného vrcholu do druhého"
- ▶ Formálne: $E = (v_1, v_2)$ (usporiadaná dvojica)

Grafové algoritmy

Prehľadávanie grafu:

BFS	$\Theta(n + m)$
DFS	$\Theta(n + m)$

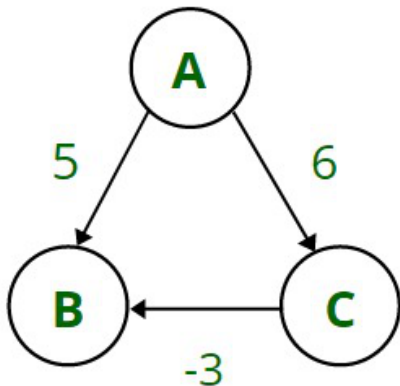
Hľadanie najkratších ciest:

algo	obmedzenie	výhoda	výsledok
Dijkstra	len + hrany	rýchly	len z 1 vrchola
Floyd-Warshall	nie – cykly	aj – hrany	\forall dvojice vrcholov
Bellmann-Ford	nie – cykly	aj – hrany	len z 1 vrchola

Hľadanie najlacnejšej kostry + UNION FIND

- ▶ Kruskalov
- ▶ Primov

Čo sa na Dijkstre pokazí?



Ako by behal FW ?

1. Odsimulujme na rovnakom grafe (ale neorientovanom).
2. Čo by sa zmenilo, ak by sme váhu B-C zmenili na -12?

Belman-Ford algoritmus

- ▶ Hľadá najkratšie cesty z vrcholu S do všetkých ostatných. Podobný ako dijkstra, ale skúša postupne vsúvať všetky hrany.
- ▶ $O(m \cdot n)$
- ▶ BF algo ako dynamické programovanie. Podproblém : $B[t,k]$ - najkratšia cesta z s do t , ktorá môže mať najviac k medzibodov
- ▶ Ako by sme detekovali/hľadali negatívne cykly?

Modifikovanie grafu so zápornými hranami

- ▶ Pomohlo by nám ak by sme si hrany upravili tak, že k nim pripočítame konštantu aby neboli záporné?

Modifikovanie grafu so zápornými hranami

- ▶ Pomohlo by nám ak by sme si hrany upravili tak, že k nim pripočítame konštantu aby neboli záporné?
- ▶ Čo sa stane s dĺžkou ciest, ak vrcholom priradíme (ľubovoľné) hodnoty $h[v]$ a $w'(u,v)=w(u,v)+h[u]-h[v]$?
- ▶ Ako zvoliť hodnoty $h[v]$, aby sme prevážovali záporné hrany na nezáporné?

Johnsonov algoritmus

- ▶ všetky dvojice v $O(n^2 \cdot \log n + nm)$, bez negat. cyklov

Johnsonov algoritmus

- ▶ všetky dvojice v $O(n^2 \cdot \log n + nm)$, bez negat. cyklov
- ▶ Bellman-Ford + Dijkstra

Johnsonov algoritmus

- ▶ všetky dvojice v $O(n^2 \cdot \log n + nm)$, bez negat. cyklov
- ▶ Bellman-Ford + Dijkstra
- ▶ Na nájdenie $h[]$ values použijeme BF = pridanie vrchola S , najkr. cieta z S do u bude $h[u]$

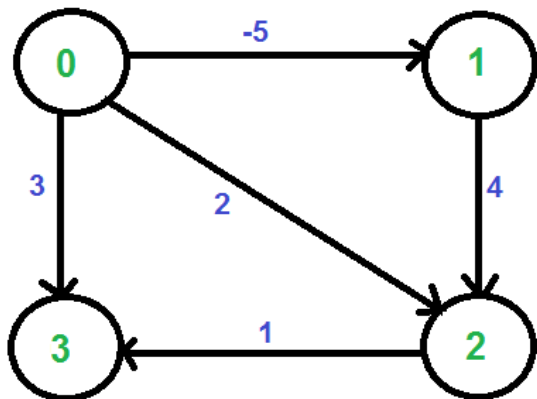
Johnsonov algoritmus

- ▶ všetky dvojice v $O(n^2 \cdot \log n + nm)$, bez negat. cyklov
- ▶ Bellman-Ford + Dijkstra
- ▶ Na nájdenie $h[]$ values použijeme BF = pridanie vrchola S , najkr. cieta z S do u bude $h[u]$
- ▶ Preváhuje graf $w'(u, v) = w(u, v) + h[u] - h[v]$, teraz budú všetky hrany nezáporné

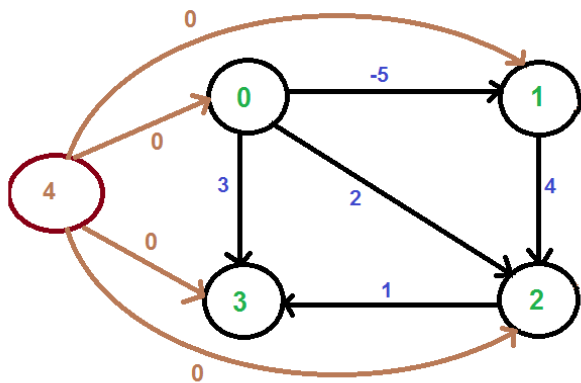
Johnsonov algoritmus

- ▶ všetky dvojice v $O(n^2 \cdot \log n + nm)$, bez negat. cyklov
- ▶ Bellman-Ford + Dijkstra
- ▶ Na nájdenie $h[]$ values použijeme BF = pridanie vrchola S , najkr. cieta z S do u bude $h[u]$
- ▶ Preváhuje graf $w'(u, v) = w(u, v) + h[u] - h[v]$, teraz budú všetky hrany nezáporné
- ▶ Vymažeme S , pustíme dijkstru z každého vrchola

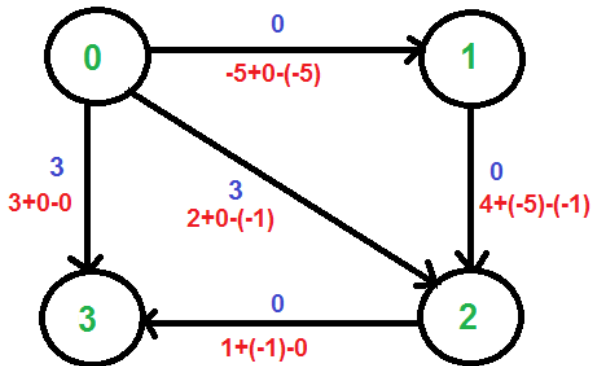
Johnsonov algoritmus



Johnsonov algoritmus



Johnsonov algoritmus



Distances from 4 to 0, 1, 2 and 3 are 0, -5, -1 and 0 respectively.

Úloha - jazdenie po meste

- ▶ najviac zdržuje státie na semafore
- ▶ na semafore treba stáť, ak pokračujeme rovno alebo doľava
- ▶ doprava môžeme odbočiť hneď (je tam odbočovací šípka)
- ▶ nájdite najrýchlejšiu cestu z bodu A do B, najrýchlejšia = najmenej stáť na semafore