Dijkstrův alg.

Časová složitost:

- jeden cyklus jde přes všechny nenavštívené vrcholy
  - druhý z takových vrcholů hledá všechny cesty.

To celé má složitost $O(n^2)$
- vnější hledání proběhne nejvýš n-krát
- uvnitř pak lineárně procházím ostatní vrcholy a hrany.

Postup algoritmu:

1) $\forall_v \; h(v) \Leftarrow +\infty, \; s(v) \Leftarrow$ nenavštívený
2) $h(u) \Leftarrow 0, \; s(u) \Leftarrow$ otevřený
3) $\quad v \Leftarrow$ otevřený s min. $h(v)$
4) $\quad$ Pro všechny hrany vw:
5) $\quad\quad$ Pokud $h(v) + \ell(v,w) < h(w)$
6) $\quad\quad\quad h(w) \Leftarrow h(v) + \ell(v,w)$
7) $\quad\quad\quad s(w) \Leftarrow$ otevřený
8) $\quad s(v) \Leftarrow$ zavřený

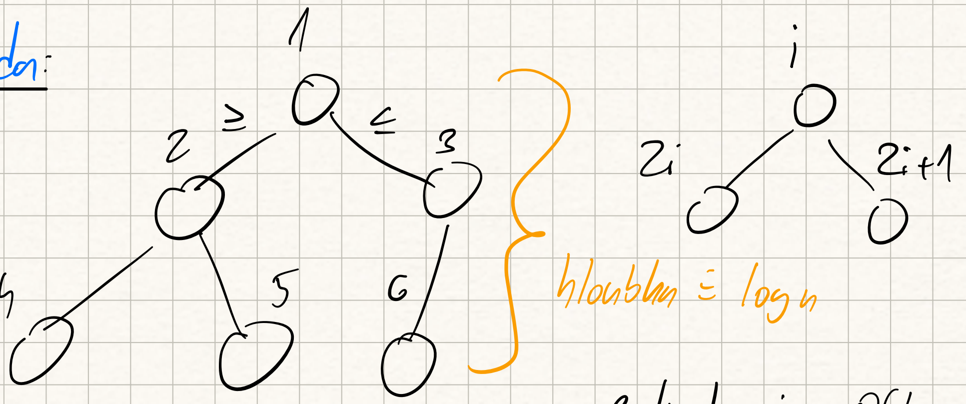
Chceme DS pro dynamické minimum

- pamatuje si otevřené vrcholy a jejich ohodnocení
- Operace:
  - extract min. – najde a smaže vrchol s min. ohodnocení $\leq n$ (vrcholů)
  - insert – vloží nový vrchol $\leq n$ (vrcholů)
  - Decrease – sníží ohodnocení vrcholu $\leq m$ (hran)

$$O(n \cdot T_{insert} + n \cdot T_{extr. min} + m \cdot T_{decrease})$$

Pole: $(n \cdot 1 + n \cdot n + m \cdot 1) = O(n^2)$ → nic nenušetříme

Binární halda:



hloubka ≈ log n

vlevo menší, vpravo větší

prvek je sice nahoře, ale pak musím probublat novou hodnotu.

extract min $= O(\log n)$
Insert $= O(\log n)$
Decrease $= O(\log n)$

- Zároveň se ale musí uchovávat, kde se jednotlivé prvky nachází, protože halda není vyhledávací strom.

Složitost?

$$O(n \cdot \log n + n \cdot \log n + m \cdot \log n)$$

$$O \dots (n \log n) \text{ až } (n^2 \log n)$$

Je to pomalé pro husté grafy!

→ Existuje Fib. halda:
Insert, Decrease $\sim O(1)$
Extract $\sim O(\log n)$
⇓
Dijkstra $\sim O(n \log n + m)$

Relaxace:

- Snažím se $h(w)$ snížit na $h(v) + \ell(v,w)$

Jsou potřeba stavy, abych se nezacyklil:

- otevřený: od předchozí relaxace se hodnota změnila → má smysl znovu relaxovat.

Obecný relaxační algoritmus:

1. $\forall v\ h(v) \leftarrow +\infty$, $s(v) \leftarrow$ nenavštívený, $h(u) \leftarrow 0$, $s(u) \leftarrow$ otevřený
2. Dokud $\exists v$ otevřený: relaxuj v. Dijkstra: vybere v otevřený s min. $h(v)$

Pro grafy, které mohou mít záporné hrany:

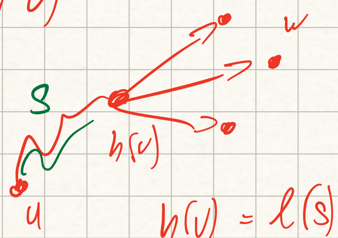
Invariant O: —> invariant popisuje nějaké věci algoritmu, co se nemění

1) $\forall v$ $h(v)$ nikdy neroste
2) pokud je $h(v)$ konečné, je rovno nějakému u,v sledu.

① - relaxace jenom snižuje, není tedy jak zvýšit.

② inic. OK (vrchol 0)

Relaxace:



$h(v) = \ell(s)$

$h(v) + \ell(v,w) = \ell(s')$; $s' = s + (v,w)$

tedy je to délka nějakého sledu $\boxed{V}$

Lemma o dosažitelnosti:

Pokud se algoritmus zastaví, $\forall v \in V$: v je dosažitelný z u, v je uzavřený, $h(v)$ je konečné.

viz korektnost BFS/DFS

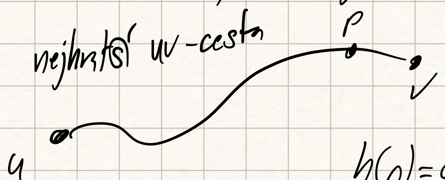
Jakmile mám něco dosažitelného a program skončí, musí to být zavřené, jinak bych nezavřel nic jiného v grafu.

Lemma o vzdálenosti:

Pokud se algoritmus zastaví, $\forall v \in V$: $h(v) = d(u,v)$, kde u je začátek

- pokud není v dosažitelný, pak $h(v) = +\infty$, $|d| = +\infty$
- Pokud je dosažitelný, je i vzdálenost konečná.
  - díky Inv. O: $h(v) \geq d(u,v)$
    - Sporem: Ukážu $h(v) < d(u,v)$ —> v je „špatný" vrchol
      - vybereme v „špatný", t.ž. nejkratší uv-cesta má nejméně hran

$u \neq v$



v je špatný, má dobrého předchůdce p.: $h(p) = d(u,p)$

Někdy jsme toto $h(p)$ museli nastavit

– tehdy jsme p otevřeli, později museli relaxovat

– tedy projít všechny následovníky, což je i v.

– Po relaxaci: $h(v) \le \underbrace{h(p) + \ell(p,v)}_{\underbrace{d(u,p) + d(p,v)}_{d(u,v)}}$ $\boxed{↯}$ $h(v) \le d(u,v)$

Zpět k Dijkstře:

Na grafu s $\ell \ge 0$

Invariant M:

Kdykoliv je o otevřený vrchol a z zavřený, pak ① $h(z) \le h(o)$
② $h(z)$ se nemění

zavřené $\le$ otevřené

– ty zavřené jsou prostě „blíž" jak ty otevřené.

– zároveň na zavřené už nesáhneme

Indukcí podle výpočtu:

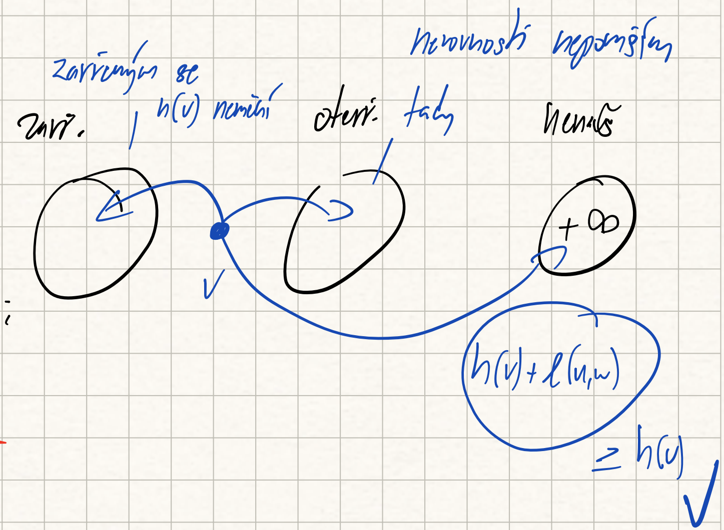
a) zpočátku ✓

b) při relaxaci:

zavř. $\le v \le$ otevř.

$h(z) \le h(v) \le h(o)$ $\boxed{✓}$

:



Věta: Dijkstrův alg. zavírá vrcholy v pořadí podle vzdálenosti, každý dosažitelný právě 1, $h(v)$ v okamžiku zavření je rovno $d(u,v)$.

Důkazy jsou poskládané výše: ① Lemma dosažitelnosti (otevření / zavření právě 1)

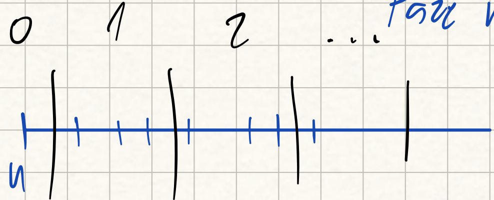
② Zavřený vrchol nemění ohodnocení – zavřený tedy jen jednou. (Lemma o vzdálenosti)

③ Od zavření se hodnocení nemění, takže ve chvíli zavírání je hodnocení finální.

Bellman-Ford alg.

- relax. alg.
- otevřené vrcholy ve frontě
  ↳ zavírá nejstarší otevřený vrchol

Bellman-Ford spočítá vzdálenosti $d(u, ?)$ v čase $O(n \cdot m)$ pro libovolný graf bez záporných cyklů.



Fáze výpočtu: $F_0 :=$ otevření u

$F_i :=$ zavírání vrcholů otevřených v F_{i-1} a otevírání jejich následovníků

Invariant: Na konci fáze F_i $\forall v \in V: h(v) \leq$ délka nejkratšího uv-sledu o max i hranách.

↳ Po nejvýše $n-1$ fázích platí: $h(v) \leq d(u,v)$

$\Rightarrow$ n-tá fáze už nic neotevře. $\Rightarrow$ $\#$fází $\leq |V(G)|$

Indukcí podle i:

a) pro $i=0$ ✓ b) Nechť jsme na konci $(i+1)$-té fáze, vrchol v, nejkratší uv-sled S

pokud S má $\leq i$ hran, už to platí už po předchozí fázi

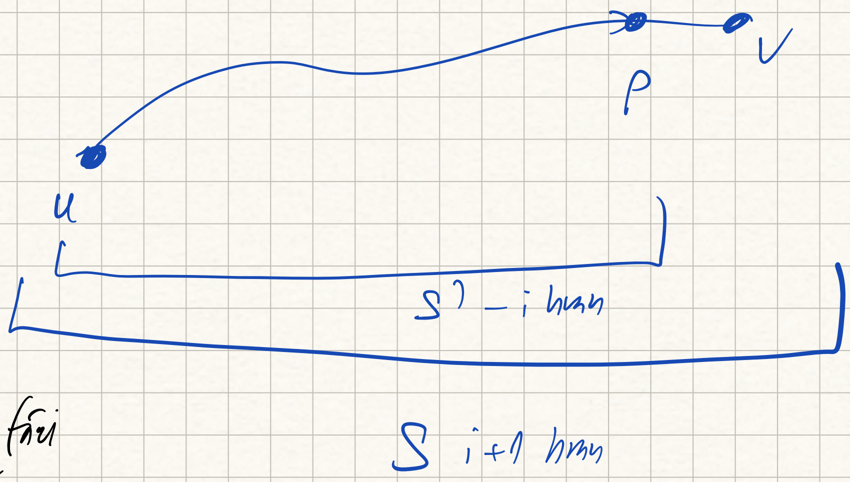
S má právě i hran:

Podle IP na konci F_i

máme $h(p) \le \ell(s')$

nastaveno nejpozději v i-té fázi.

↳ nejpozději v i+1-té fázi zavřen, tehdy relaxován.



Proto $h(v) \le \underbrace{\underbrace{h(p)}_{\le \ell(s')} + \ell(p,v)}_{\le \ell(s)}$