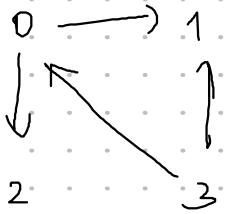
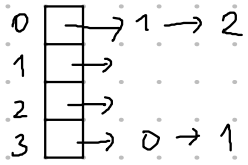
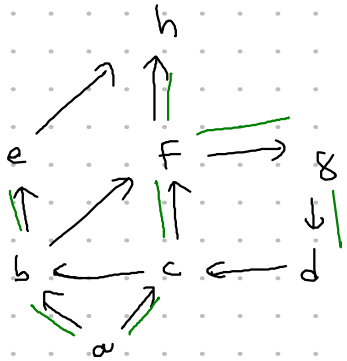




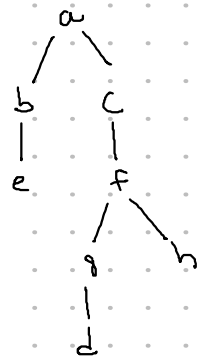
	0	1	2	3
0	0	1	1	0
1	0	0	0	0
2	0	0	0	0
3	1	1	0	0

	a	b	c	d	e	f	g	h
X	X	X	X	X	X	X	X	X

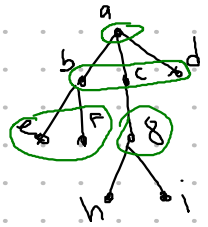


M: ~~a~~ ~~b,a~~, ~~F~~, ~~h~~, ~~g~~, ~~e~~, ~~d~~

pořadí navštěv: a, c, F, b, g, h, d, e



Fronta: $\underline{\overline{a}}_0, \underline{\overline{b,c,d}}_1, \underline{e,f,g}_2, \underline{h,i}_3$

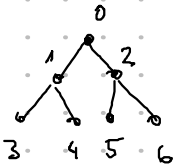


(min-)Binary heap

→ bin. strom
→ podmínka min-haap:

rodič má menší prioritu než potomek.

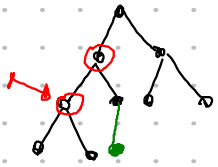
⇒ pro ∀ podstrom (kořen x), x má min prioritu z celého podstromu.



i má potomky $2i+1$, $2i+2$

j má rodiče $\lfloor j-1/2 \rfloor$

Insert ("bubble insert")



prohodím rodiče a potomka, pokud porušil min-heap podmínku (dokud ji porušuje)

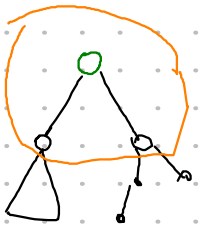
složitost: výška stromu: $\Theta(\lg n)$

decrease-key (analogicky), složitost $\Theta(\lg n)$;

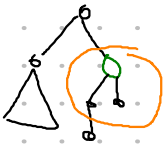
Extract-min ("bubble")

→ minimum je v kořeni

odebereme kořen a nahradíme jej posledním vrcholem



1) najdeme min. prioritu



složitost $\Theta(\lg(n))$.