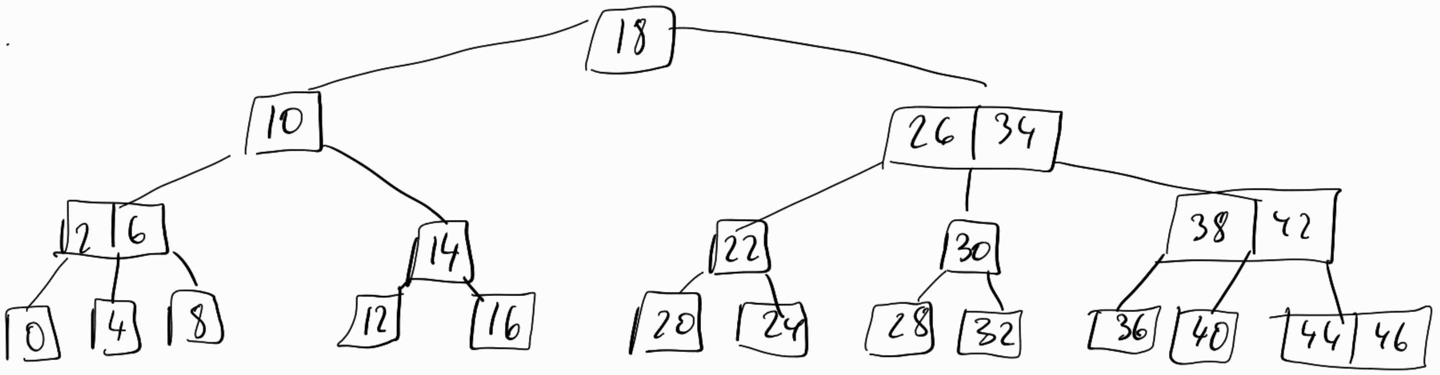
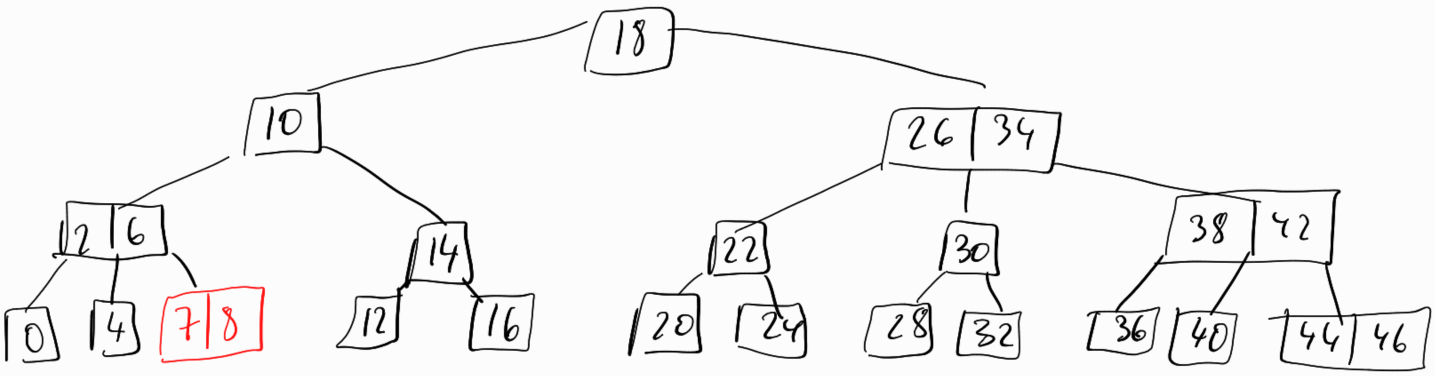
0.

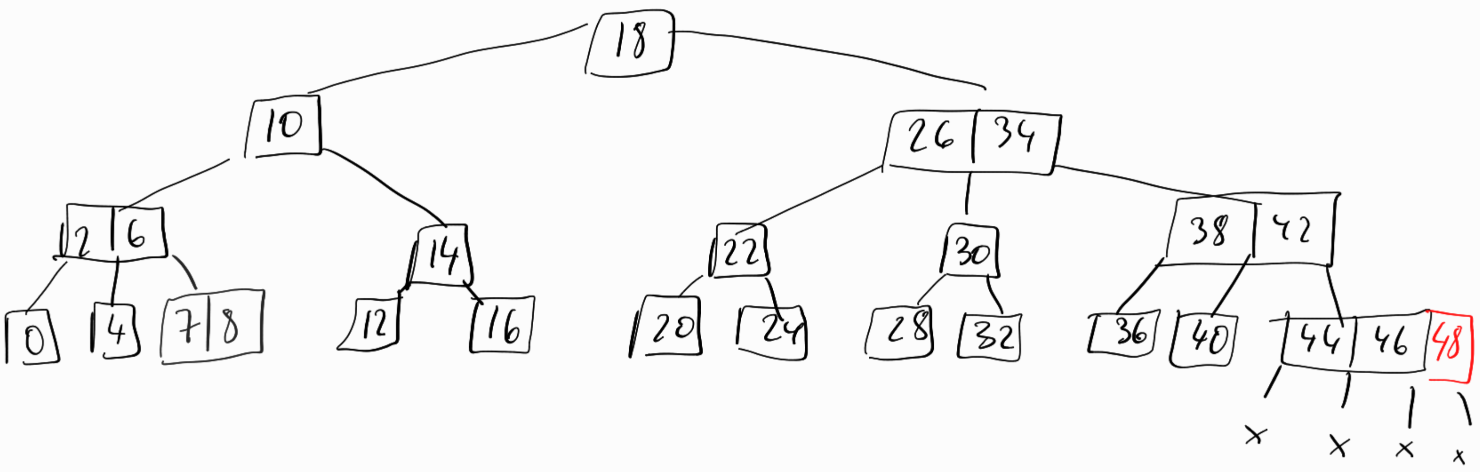


Insert (7)



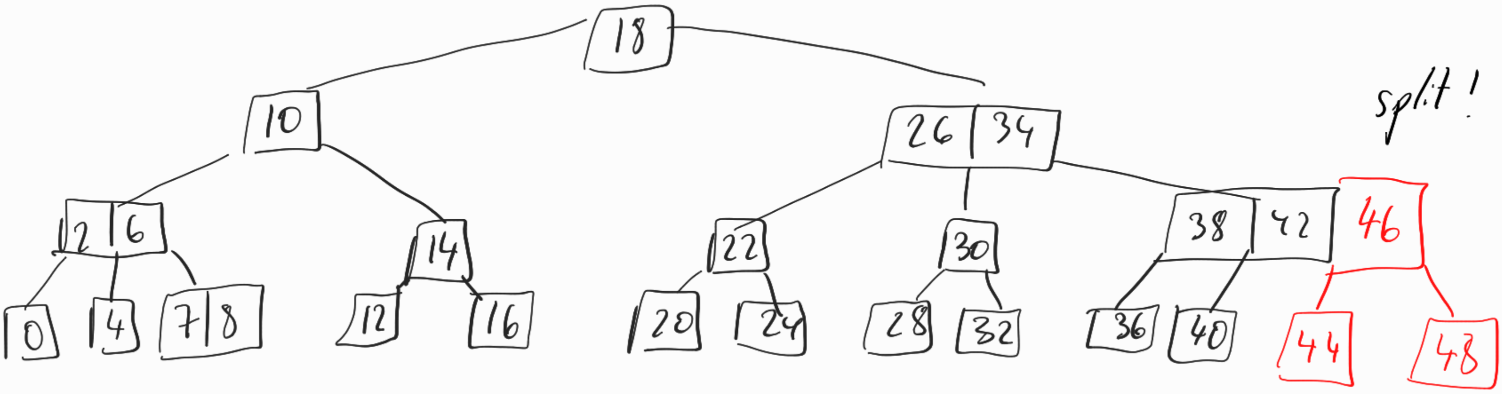
· přidalo se do rodiče listu, kam by 7 mělo jít

Insert (48)

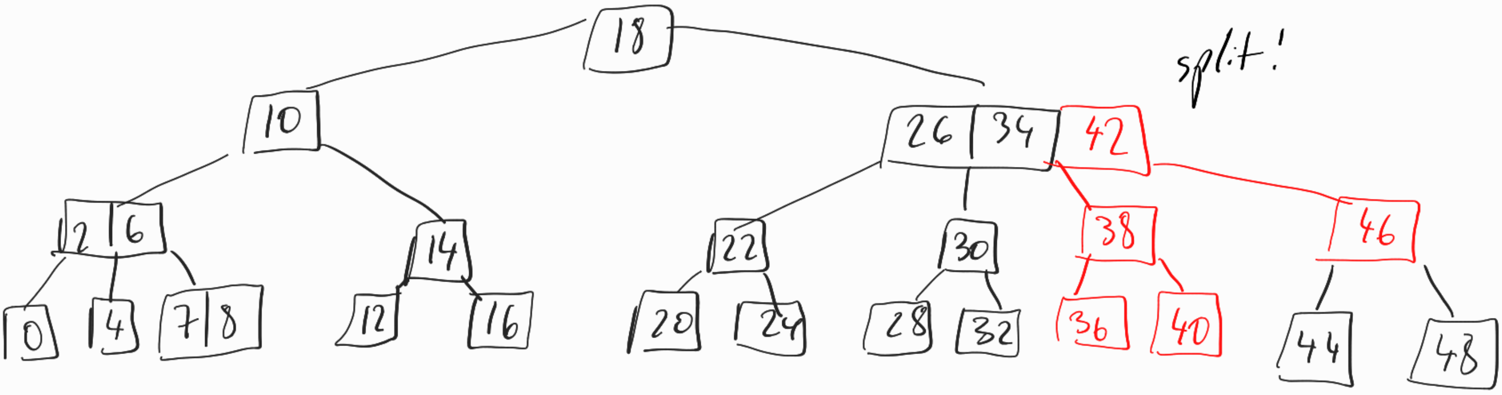


4 potomci... split!

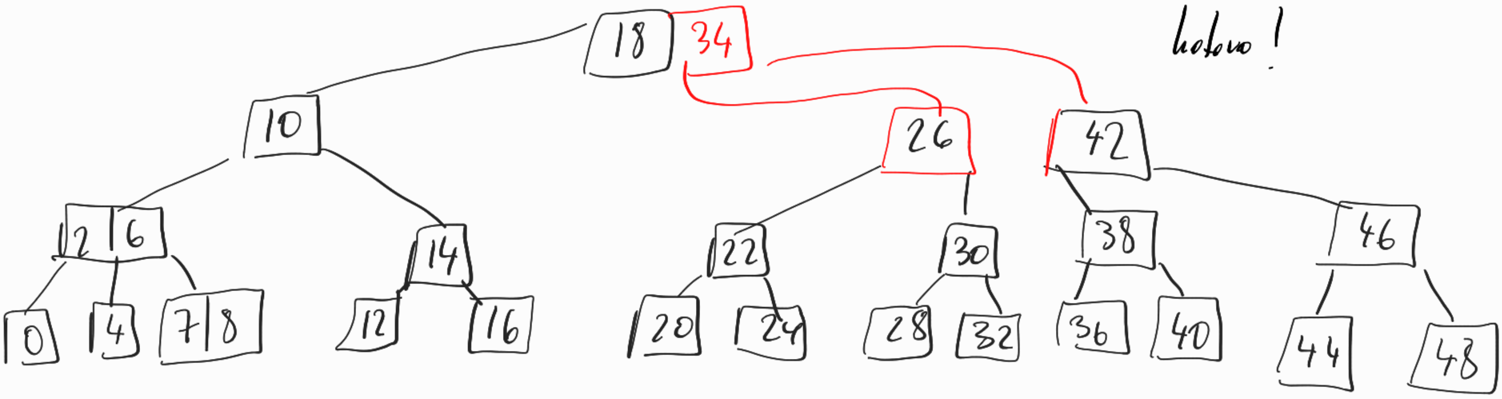
split!



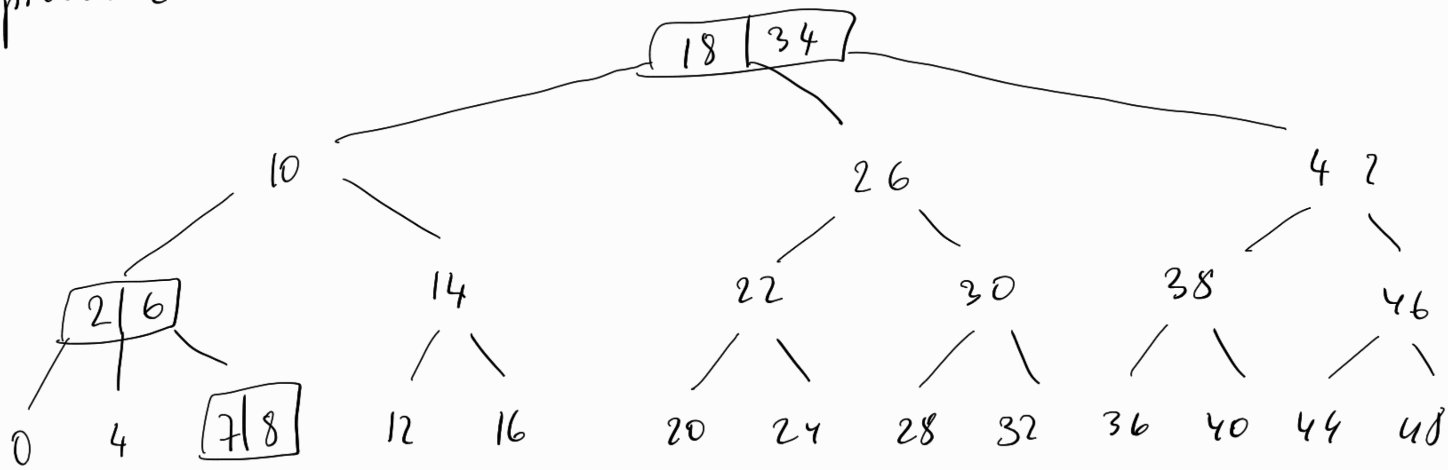
split!



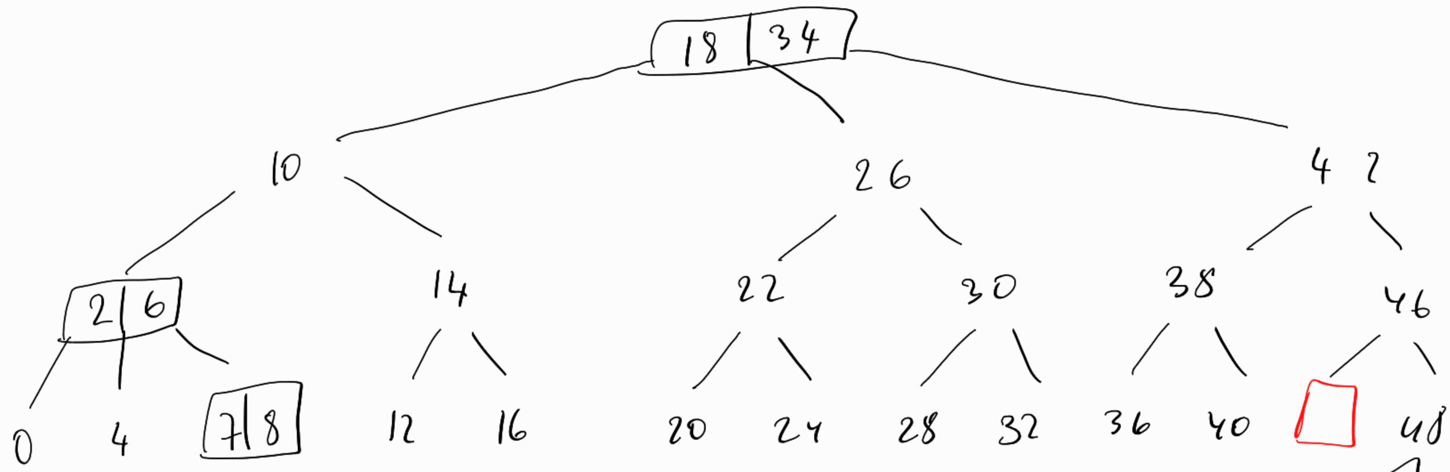
hotovo!



překreslíme

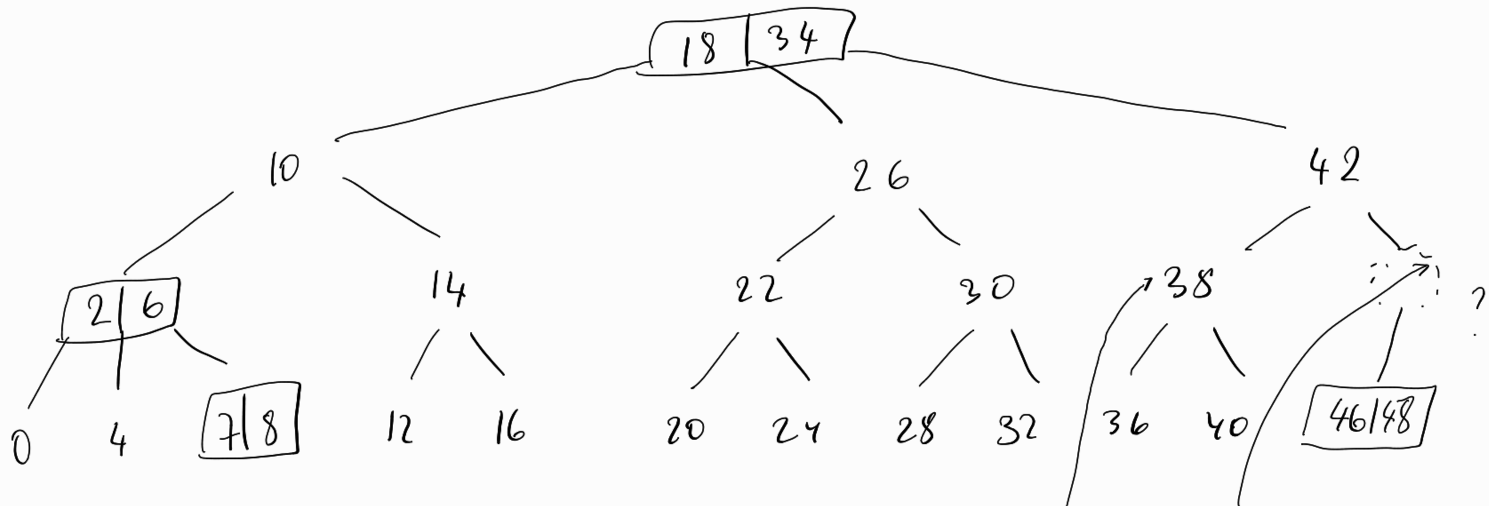


Delete (44)

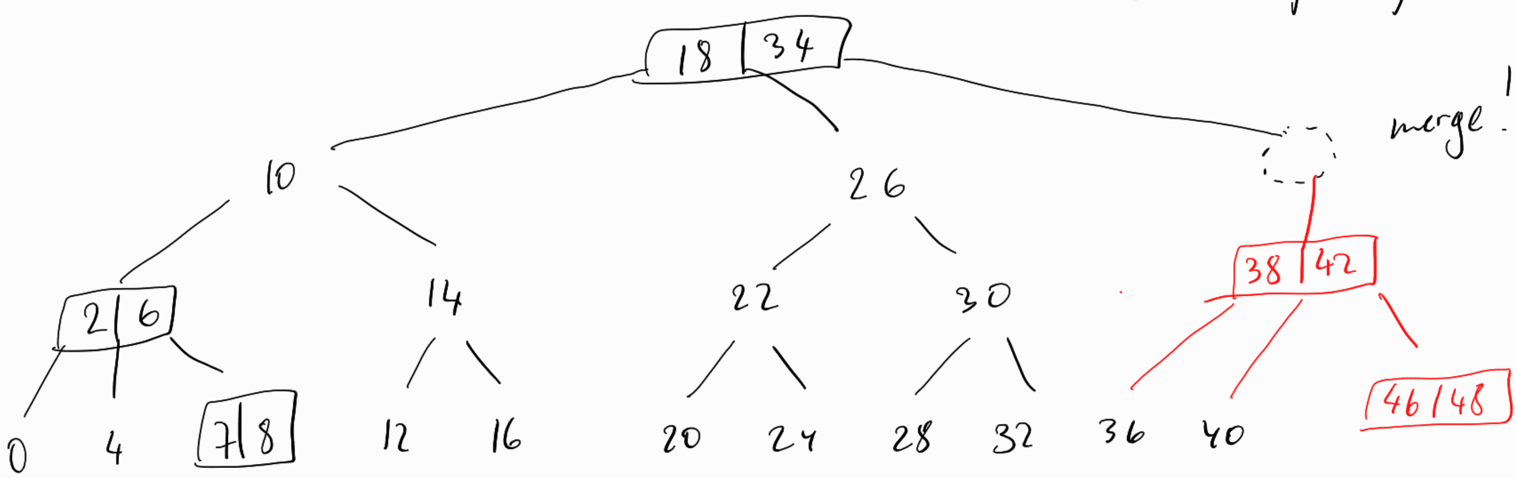


• nejde rotovat, sourozenec je na hraně (má a=2 potomků)

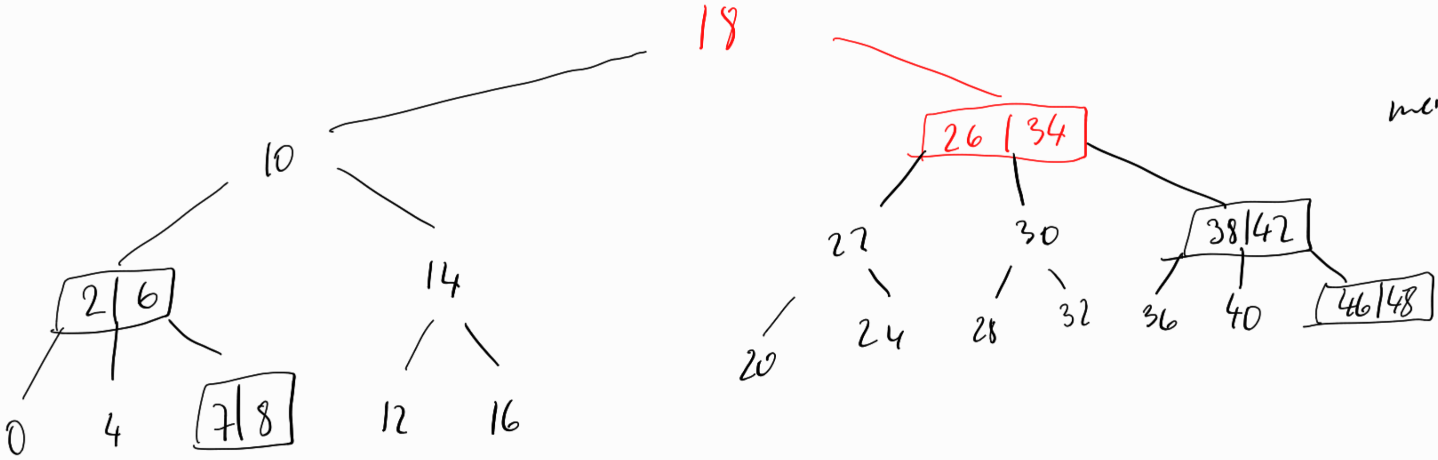
→ merge



opět merge,
sourozenec je štíhlý

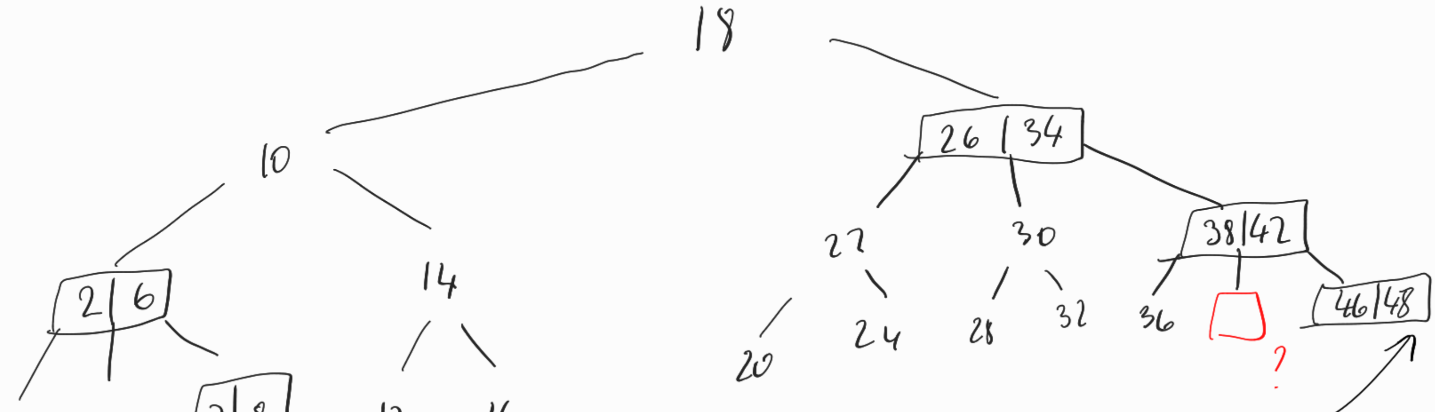


merge!

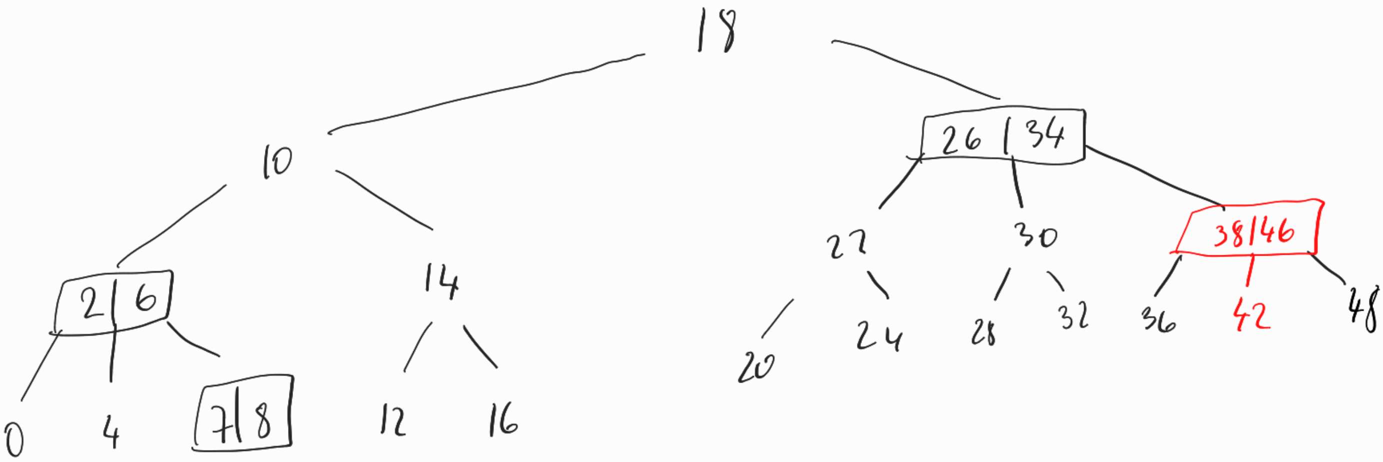


merge!

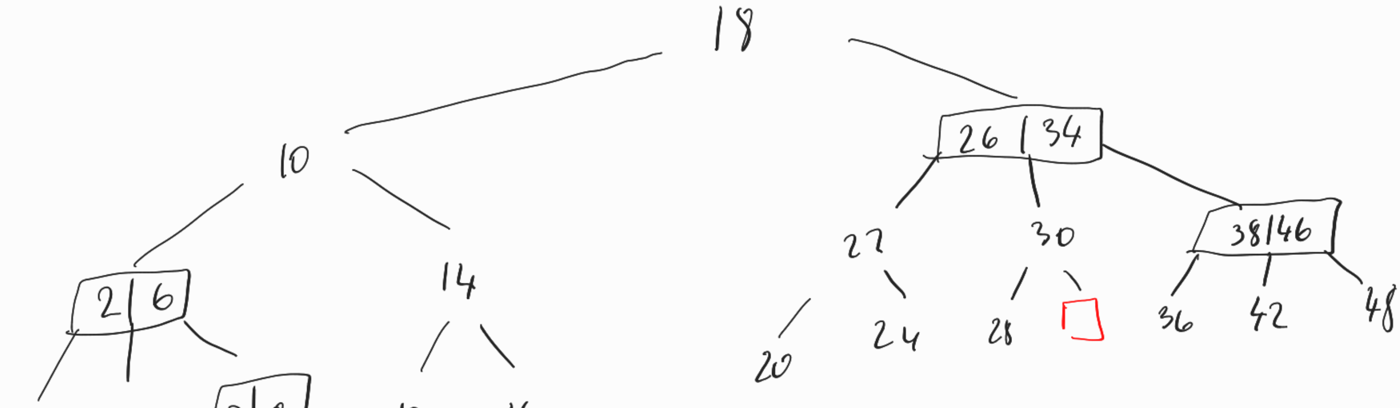
Delete 40



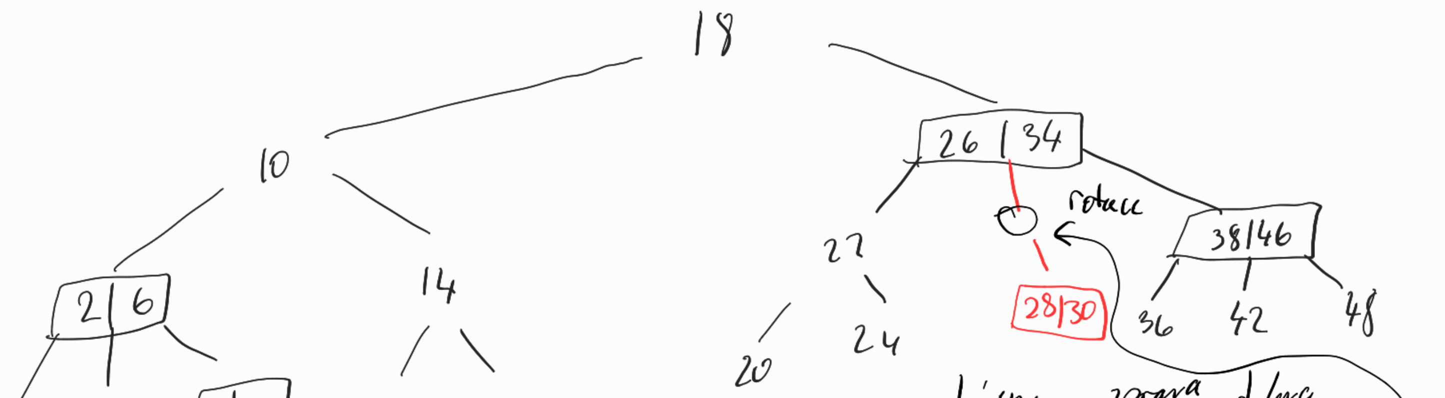
rotujeme



Delete 32

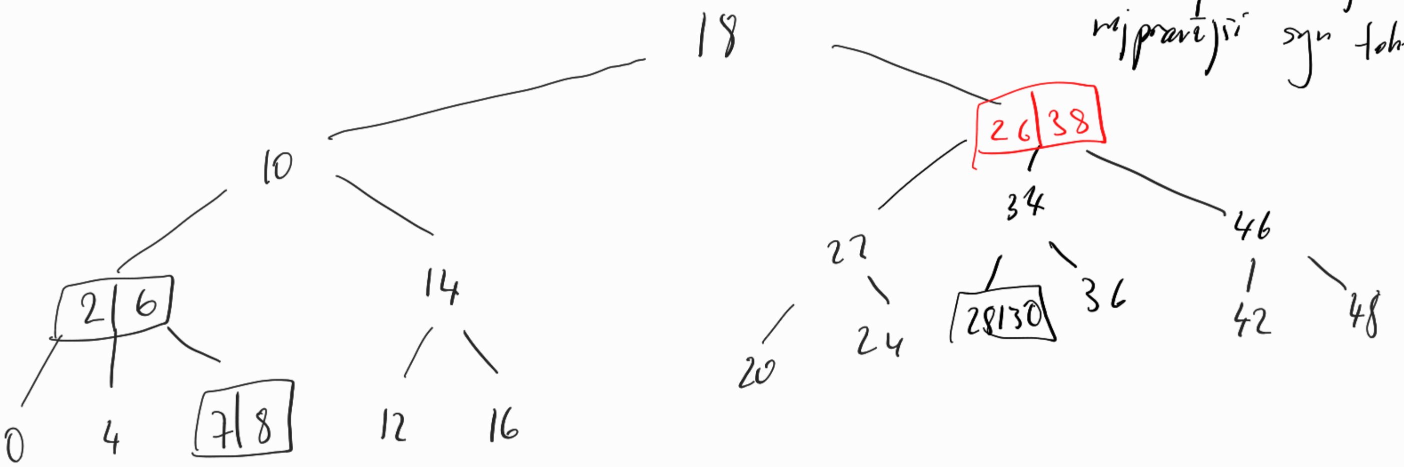


merge

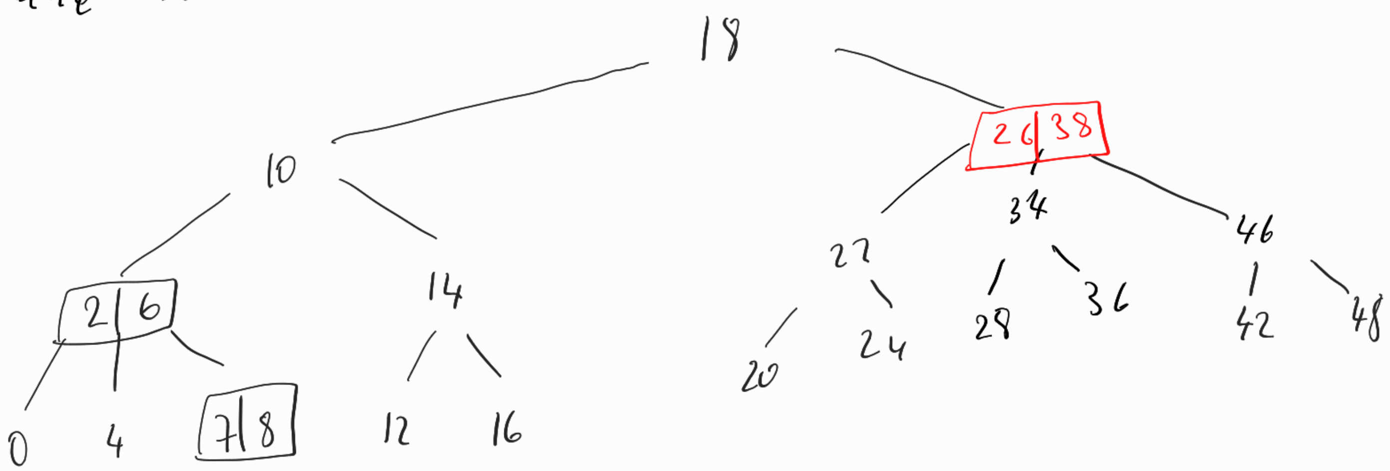


⇒ rotujeme zprava doleva

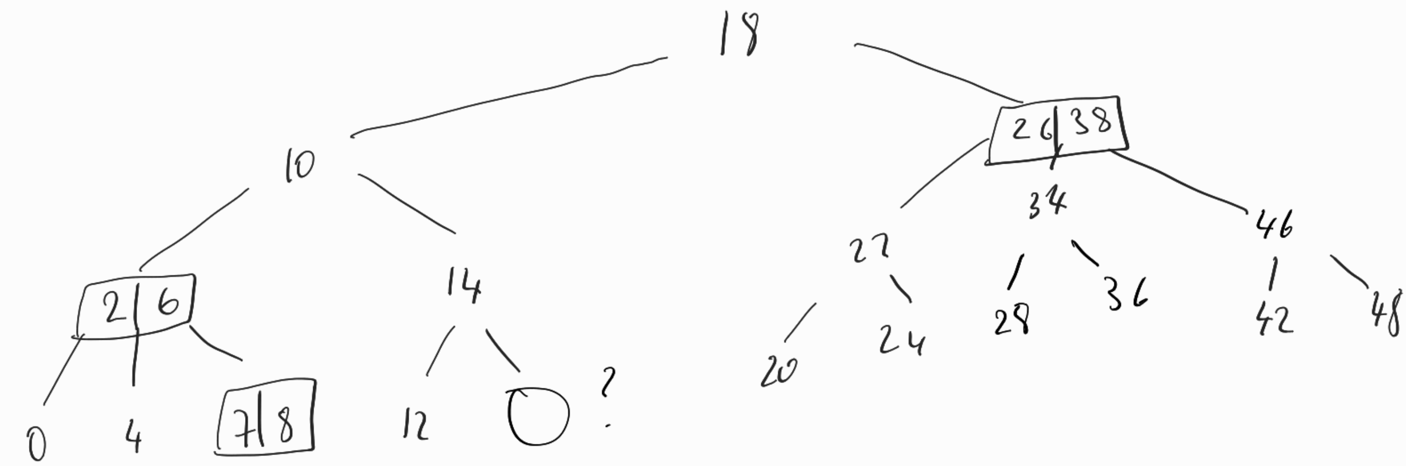
36 se přesune jako nejpravější syn tohohle

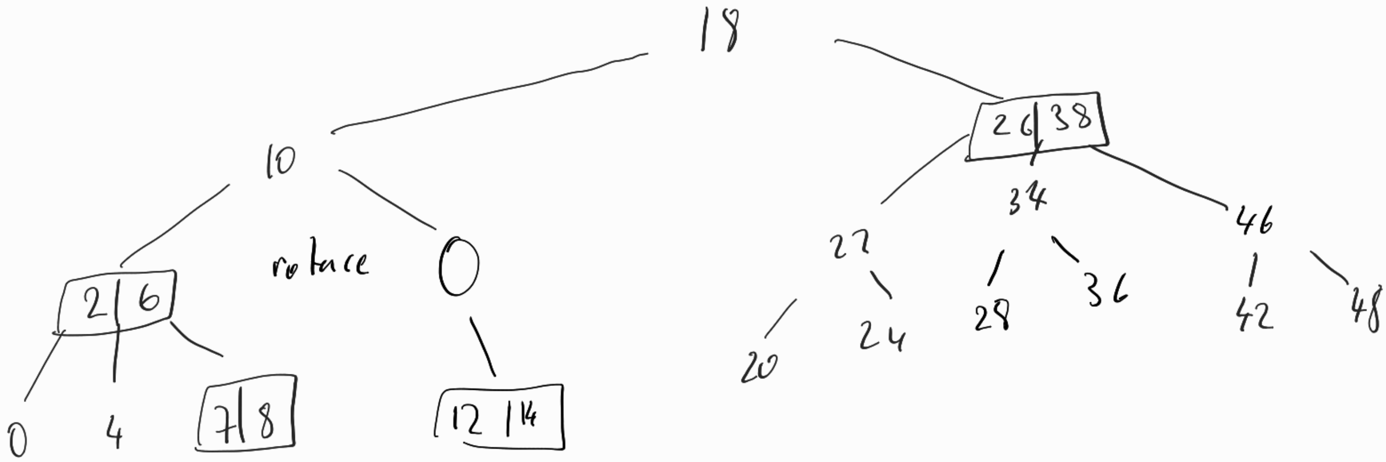


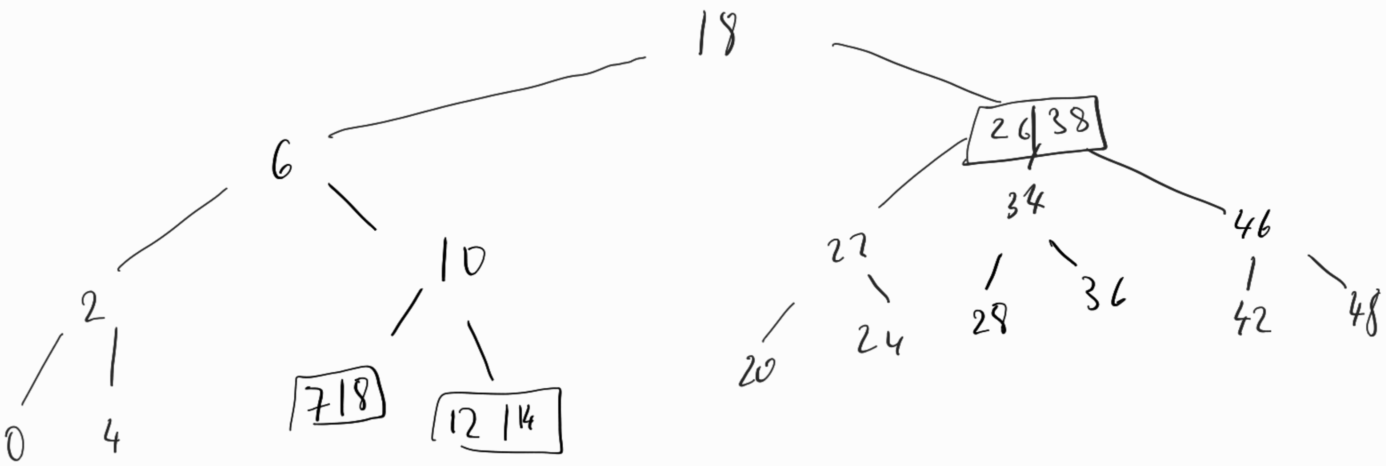
Delete 30



Delete 16



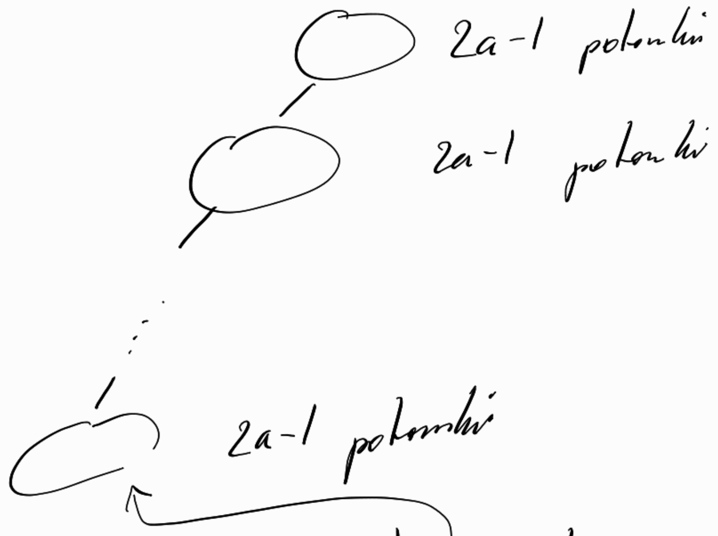




1.

- nevím, co se myslí ¯_(ツ)_/¯
- bude třeba srovnat pseudokódy

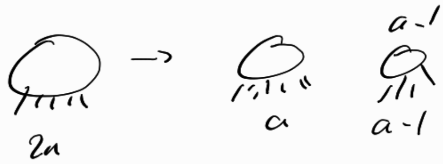
2. vyrobte strom takový, že cesta doleva jsou tučné vrcholy



- x: hodnota menší než všechny ve stromě
- insert x rozštěpí celou cestu
- delete x sloučí
- a dokola

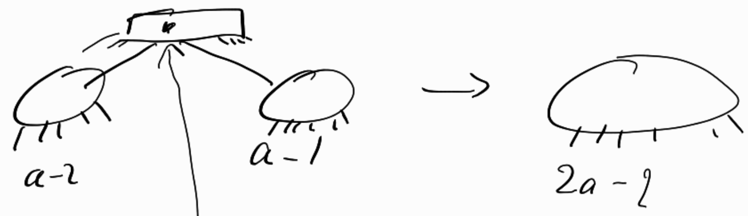
3. nechť $b \geq 2a$
- měříme # modifikovaných vrcholů na insert/delete
- najdeme potenciál Φ t.ž.
  - amort. cena štěpení a slučování je ≤ 0 vzhledem k potenciálu
  - amort. cena insert/delete je konst.
- $\Phi = \sum_{v:\text{vrchol}} f(\text{počet potomků } v)$
- kde f musí splňovat
  1. $|f(i) - f(i+1)| \leq c$ pro nějakou konst c
     - tzn. přidání klíče do vrcholu mění potenciál jen o konstantu

2. $f(2a) \geq f(a) + f(a-1) + c + 1$

- když štěpíme vrchol (ten má $\geq 2a$ potomků), tak likvidujeme $f(2a)$ potenciálu
- na rozštěpení vrcholy () potřebujeme $f(a) + f(a-1)$ potenciálu
- c za vložení prvku posledního do rodiče
- 1 ... reálná cena :)

3. $f(a-2) + f(a-1) \geq f(2a-2) + c + 1$

- když mergujeme, tak



⇝ likvidujeme $f(a-2) + f(a-1)$ potenciálu a dostáváme $f(2a-2)$ potenciálu

- c je za ukradení klíče rodiči
- 1: reálná cena :)

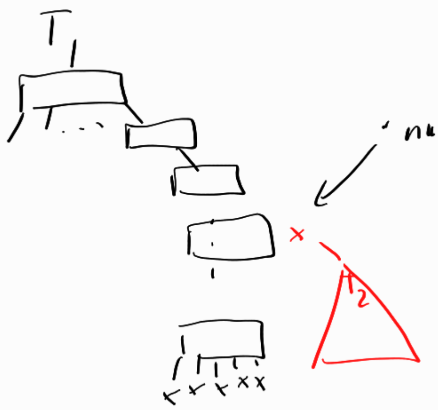
- při dostatečném přemýšlení/experimentování najdete

k	$a-2$	$a-1$	a	$\dots$	$2a-2$	$2a-1$	$2a$
$f(k)$	2	1	0	0 .. 0	0	2	4

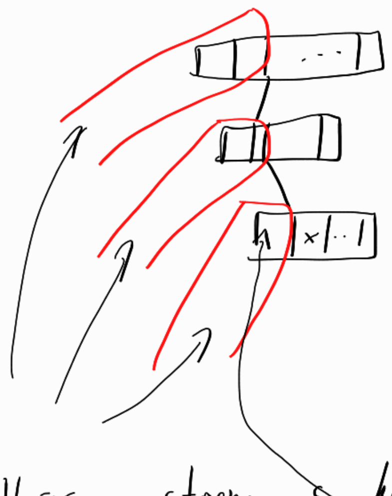
- z 2) a 3) máme zadarmo split a join, insert a delete je za c (stačí $c=2$)

4.

- zjednodušení: Merge (T_1, T_2, x) udělá merge T_1 a T_2, přičemž $\max(T_1) < \min(T_2)$ a přidá x do výsledku ($x \notin T_1 \cup T_2$, $\min T_1 < x < \max T_2$)
- búno T_1 je vyšší než T_2 (jinak lepíme T_1 do T_2)



- najdeme vrchol na pravé cestě ve výšce $h(T_1) - h(T_2)$
- do něj dáme x jako nejpravější klíč a zprava z x visí T_2
- štěpíme vrchol s x dle potřeby
- složitost $O(h(T_1) + h(T_2)) = O(\lg|T_1| + \lg|T_2|)$
- x si klidně vyrobíte smazáním min z T_2



kořen

- odkrájíme stromy s klíči menší než x a pak je pospojujeme dohromady
- tyhle mají jako kořen vrchol s aspoň 2 syny