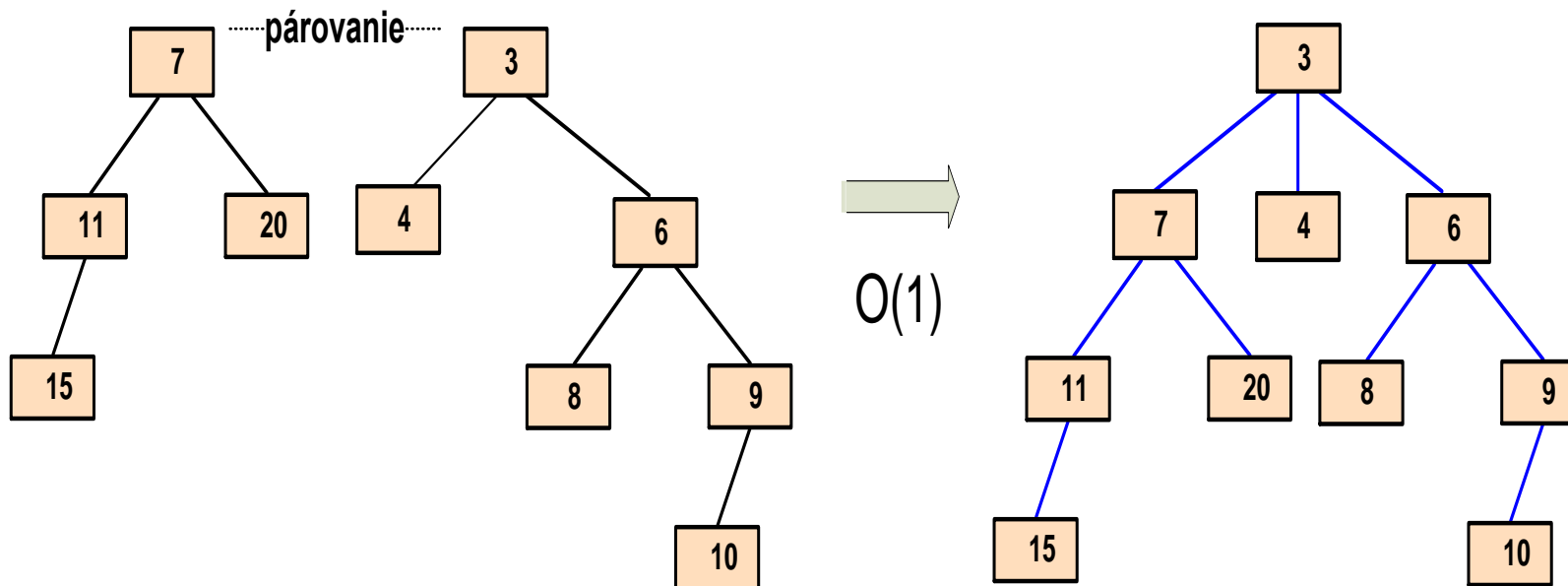


Párovacia halda

(pairing heap)

- je implementáciou prioritného frontu
- kľúč sa nazýva **priorita**
- predstavovaná **k-cestným haldovo usporiadaným stromom** (implementačne obvykle **transformovaným na binárny strom**)
- halda je strom, v ktorom pre všetky vrcholy platí, že priorita ich kľúča je vyššia ako priorita kľúčov ich priamych potomkov
- základnou operáciou je operácia **páruj**
- v príkladoch bude vyššiu prioritu predstavovať nižšie číslo
- vo všeobecnosti prioritou môže byť akýkoľvek dátový typ, je potrebné definovať, čo znamená vyššia priorita

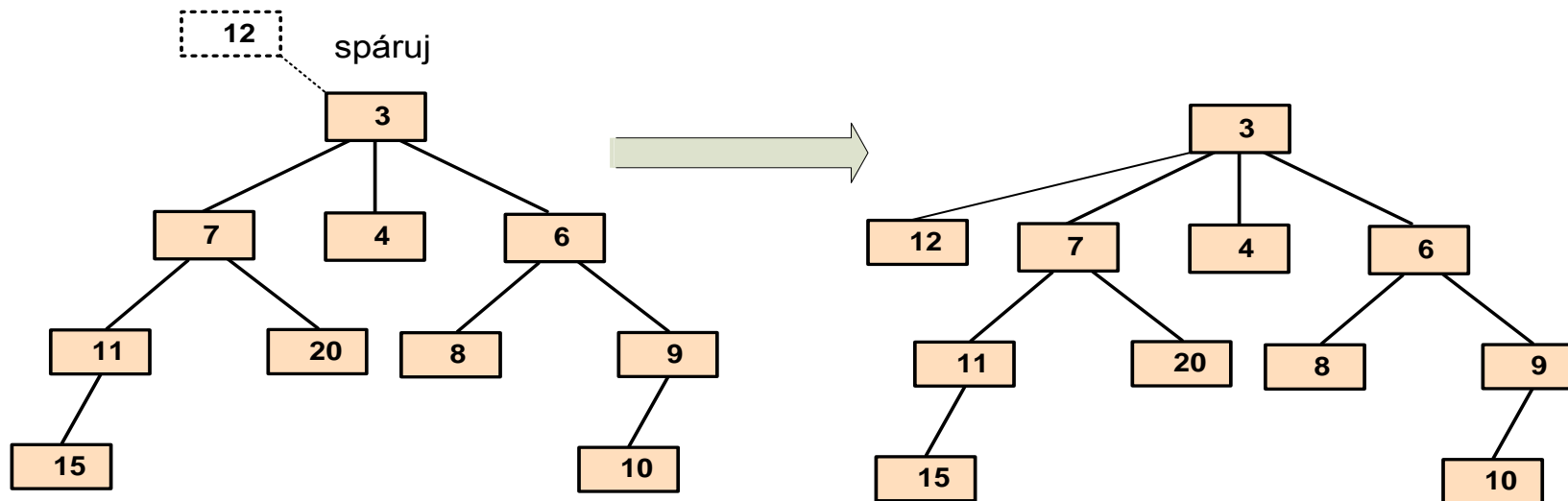
Operácie páruj na viaccestnom modeli:



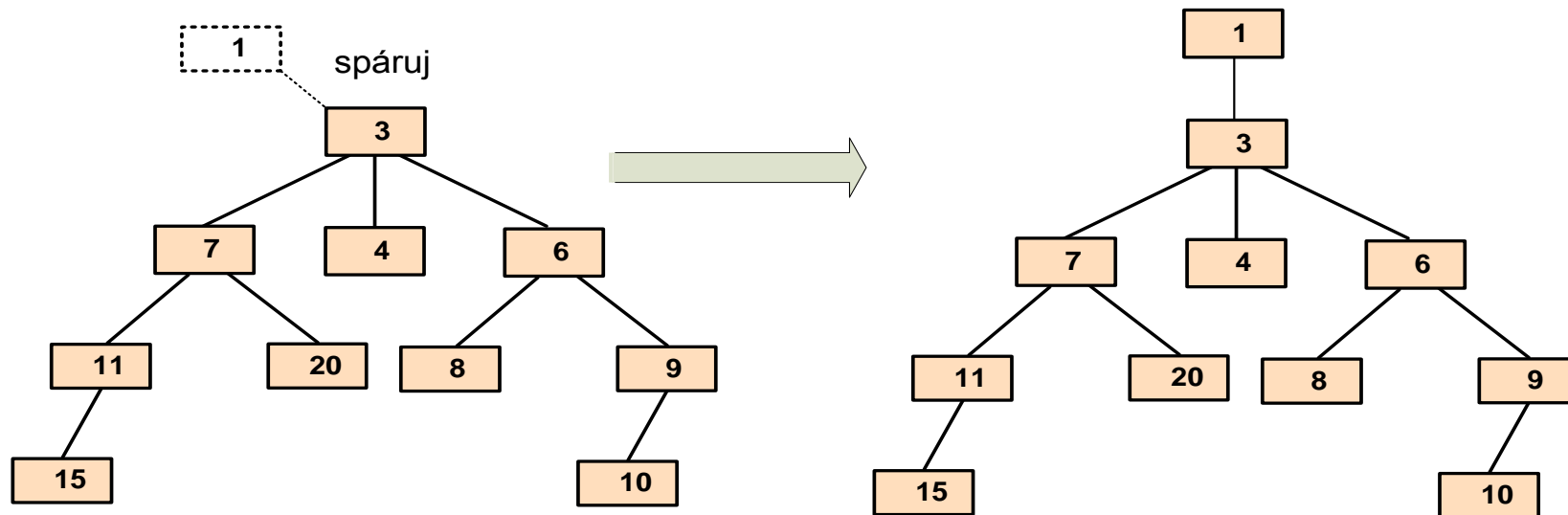
Spojenie dvoch hálď do jedinej. Porovnáme korene – koreň halďy s nižšou prioritou sa stane najľavejším synom koreňa halďy s vyššou prioritou.

Operácia párovania predstavuje vždy konštantný počet krokov a preto je jej priemerná aj najhoršia zložitosť $O(1)$.

Operácia Vlož - priemerná aj najhoršia zložitosť $O(1)$: Vkladaný prvok, ktorý je chápaný ako jednoprvková viaccestná halda sa spáruje s haldou:



(a)



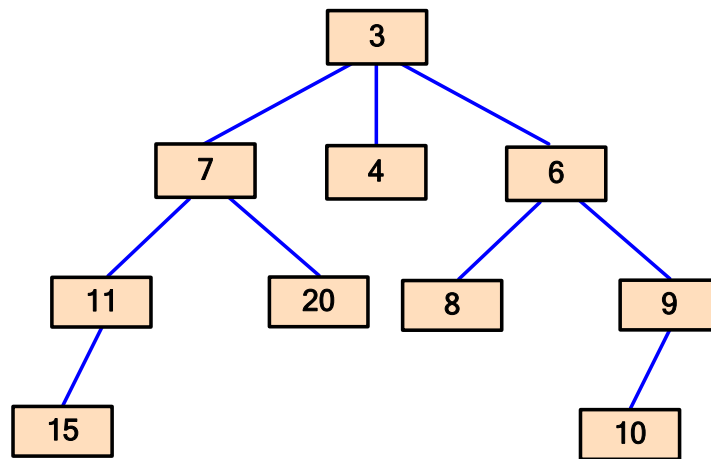
(b)

Operácia Vymaž minimum - priemerná zložitosť $O(\log n)$, najhoršia zložitosť $O(n)$,:

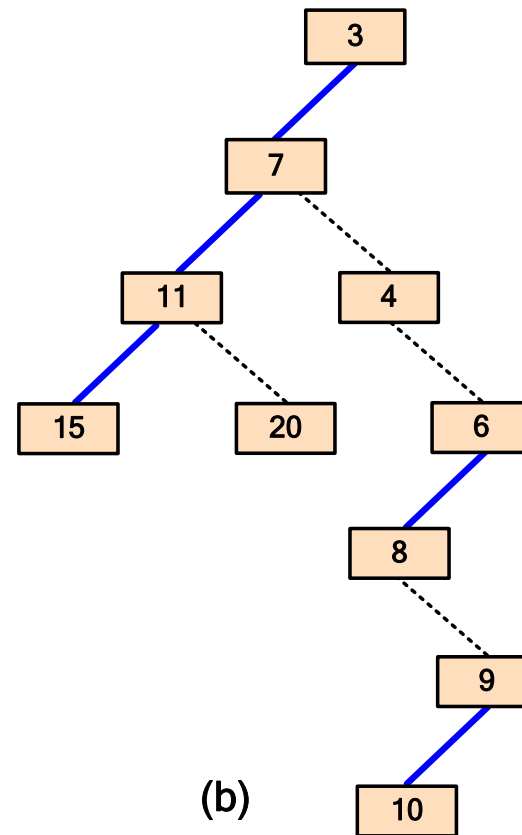
- odobratie prvku s najvyššou prioritou
- odobratie koreňa haldy, čím vznikne les stromov
- transformácia lesa postupným párovaním do jedinej haldy, rýchlosť štruktúry závisí aj od vhodného poradia párovania (vysvetlené neskôr)

Transformácia viaccestnej haldy (stromu) na binárnu

- každú haldu môžeme reprezentovať binárnym stromom, je to z hľadiska rýchlosti aplikácie lepšie, preto nie je vhodné implementovať viaccestnú párovaciu haldu
- v binárnej reprezentácii neplatia (nemôžu) platiť pravidlá haldového usporiadania prvkov, keďže pravý syn prvku v binárnej forme je jeho brat vo viaccestnej halde
- v binárnej forme platí, že všetky prvky z ľavého podstromu ľubovoľného vrcholu majú nižšiu prioritu ako tento prvok



(a)



(b)

(a) **k-cestná halda** (explicitne: synovia - počet n)

(b) **binárna halda** (explicitne: ľavý syn, pravý syn - počet 2)

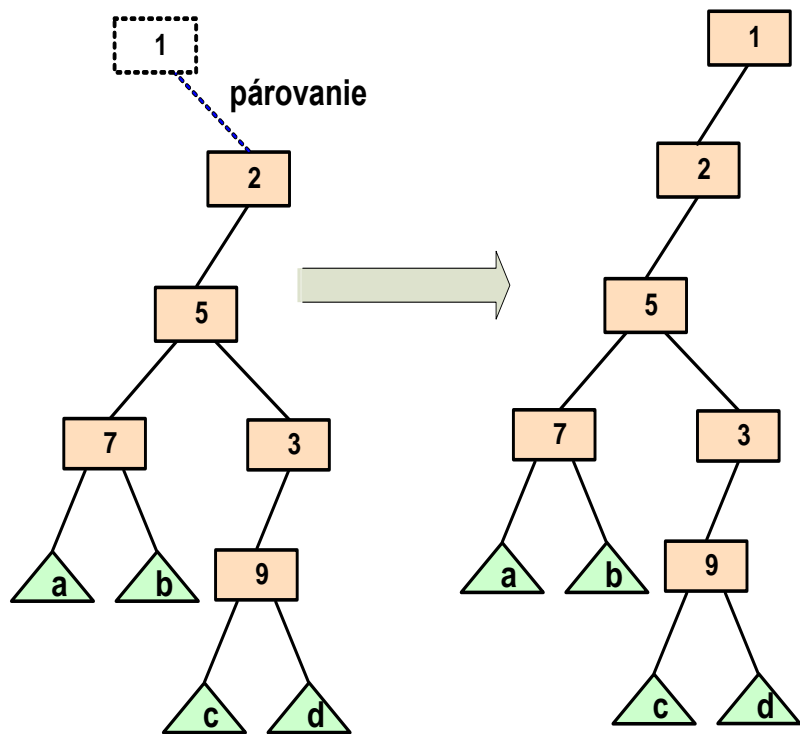
Operácie páruj na binárnom modeli:

Pravidlá:

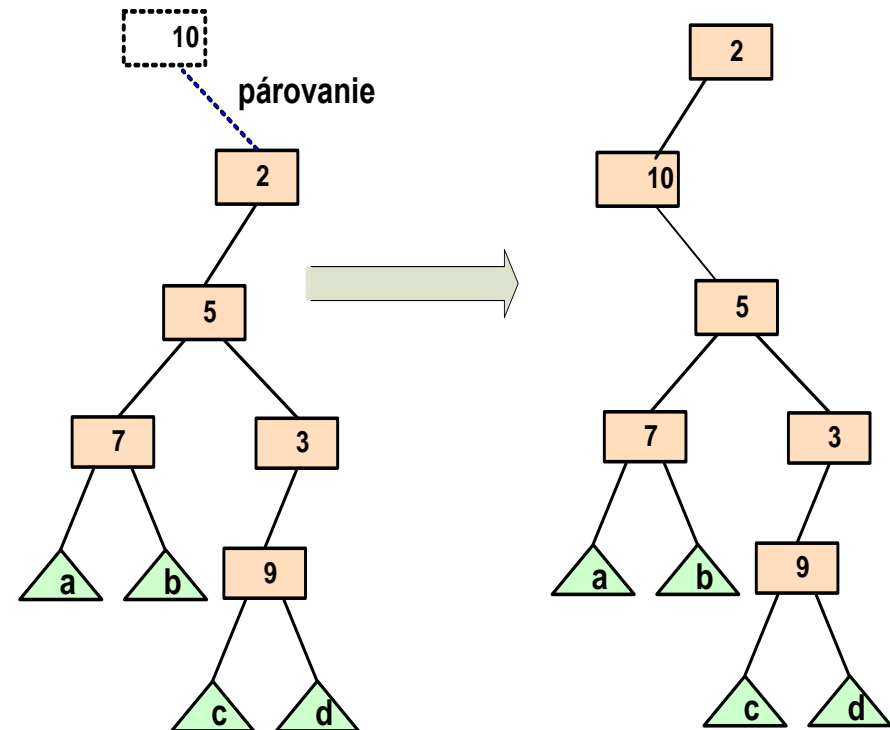
- Vrchol s väčšou hodnotou (nižšou prioritou) sa stane ľavým synom vrcholu s menšou hodnotou (vyššou prioritou)
- Doterajší ľavý syn vrcholu s menšou hodnotou (vyššou prioritou) sa stane pravým synom vrcholu s väčšou hodnotou (nižšou prioritou)

Materiál slúži výlučne pre študentov FRI ŽU, nie je dovolené ho upravovať, prípadne ďalej šíriť.

Operácia Vlož - priemerná aj najhoršia zložitosť $O(1)$:



A) vkladany menší ako koreň

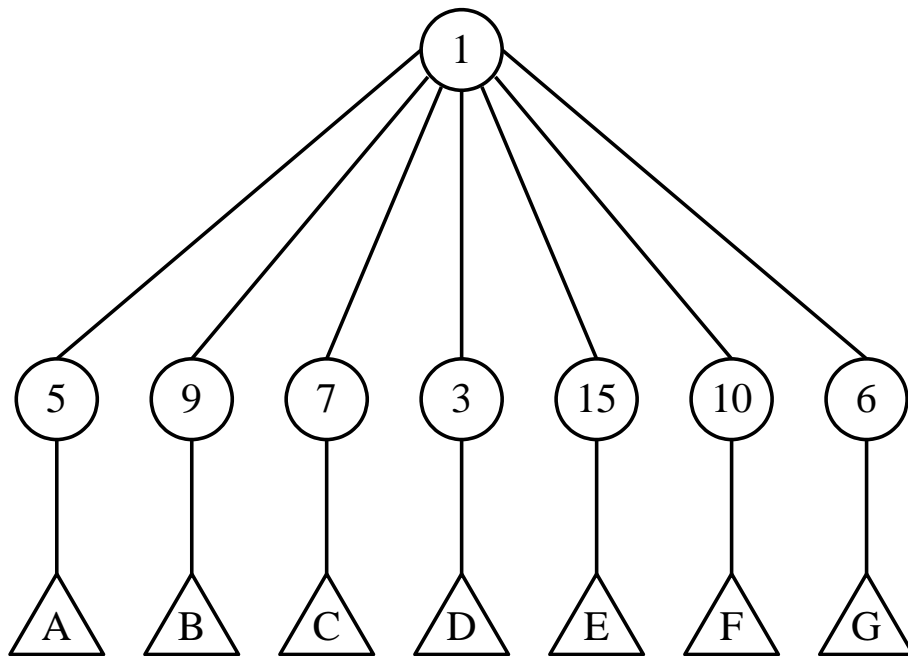


B) vkladany väčší ako koreň

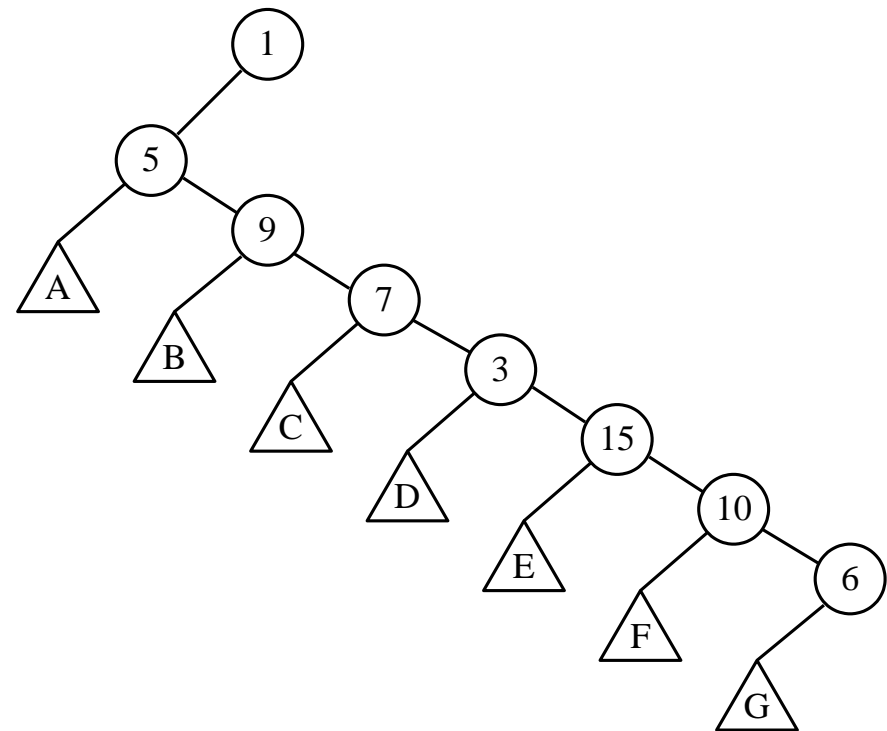
Operácia Vymaž minimum - priemerná zložitosť $O(\log n)$, najhoršia zložitosť $O(n)$:

Existujú dva vhodné postupy párovania háld po odstránení minimálneho prvku.

1. Dvojprechodová párovacia halda (Two pass pairing heap)

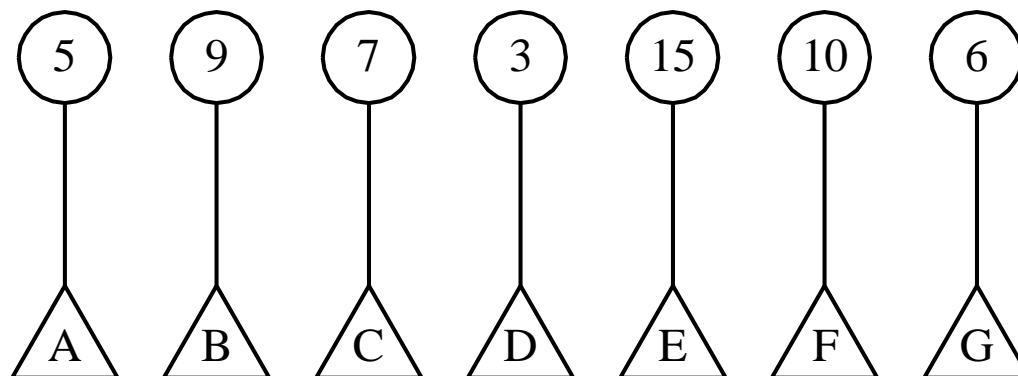


a.) viaccestný strom

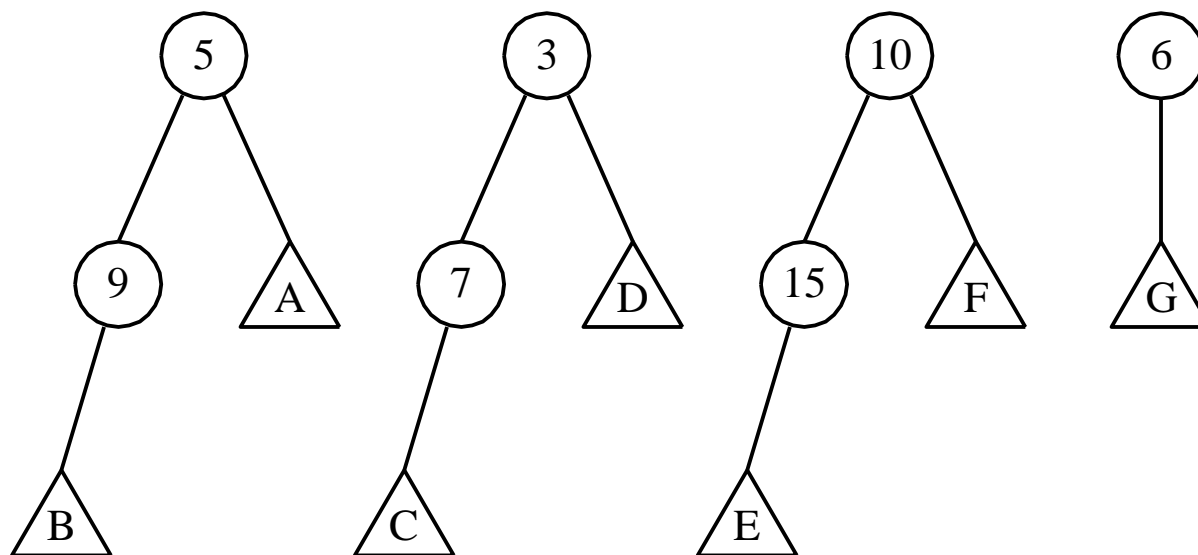


b.) binárny tvar

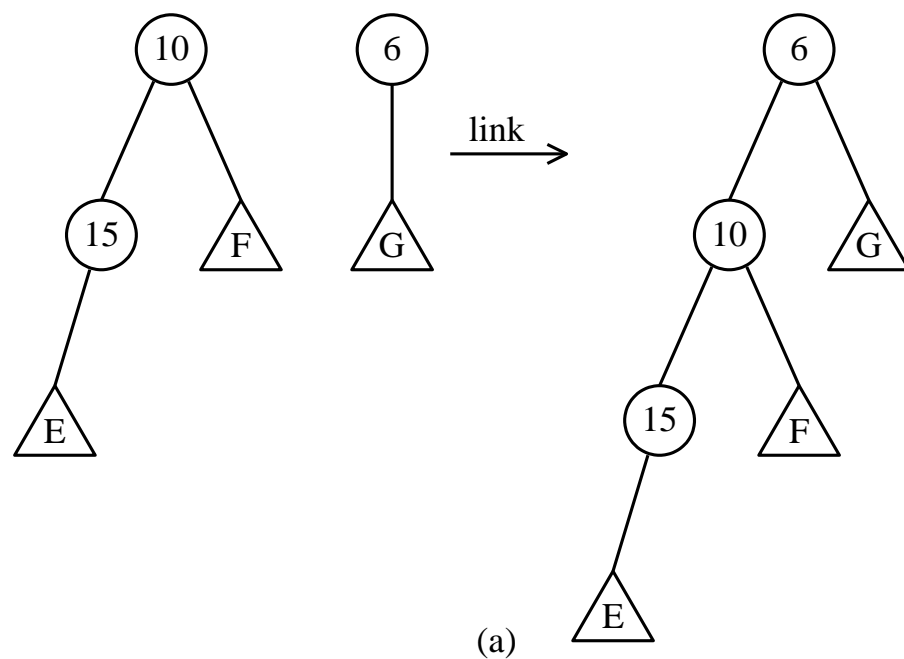
Krok 1: Výber koreňa (minimálny prvok) – výsledkom je niekoľko háld

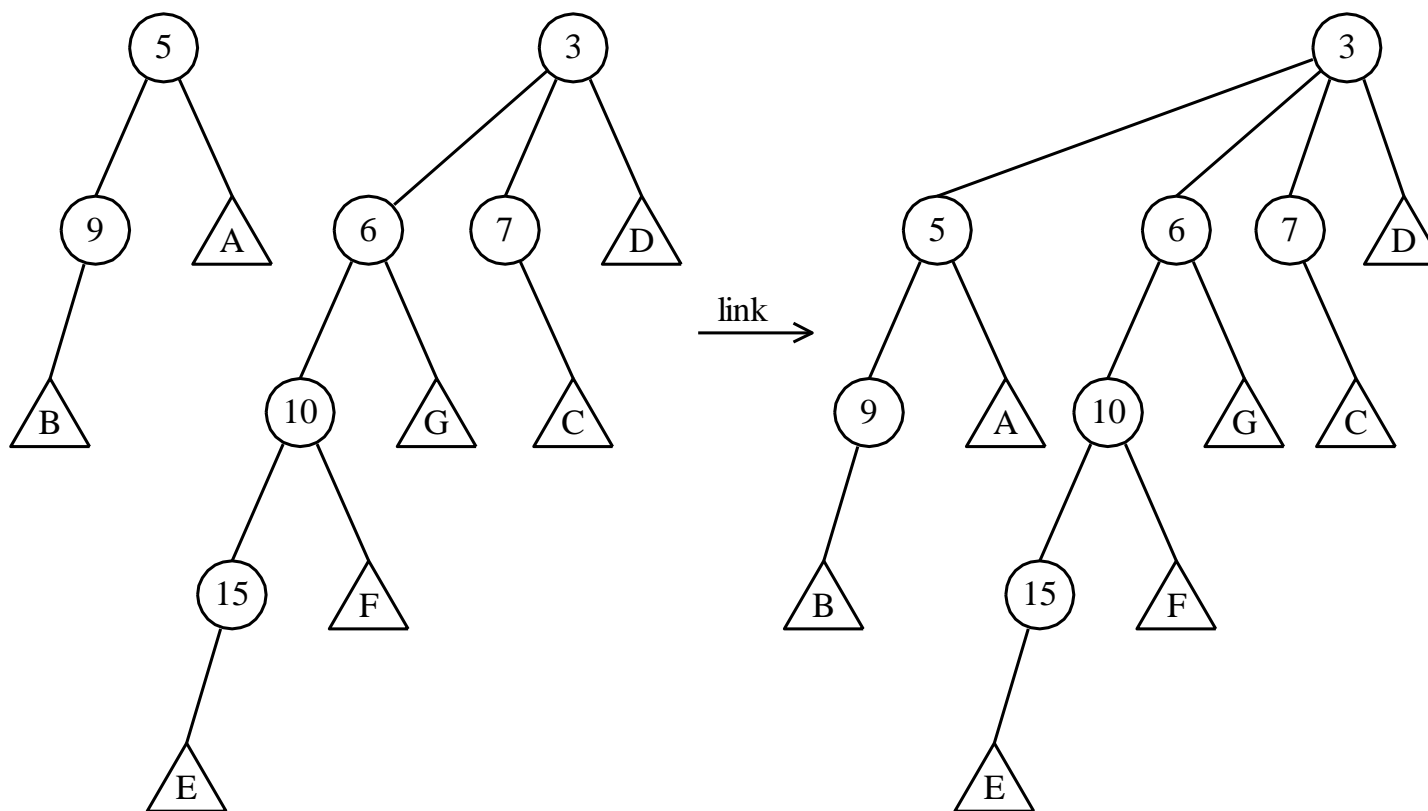


Krok 2: Postupné párovanie háld zľava doprava



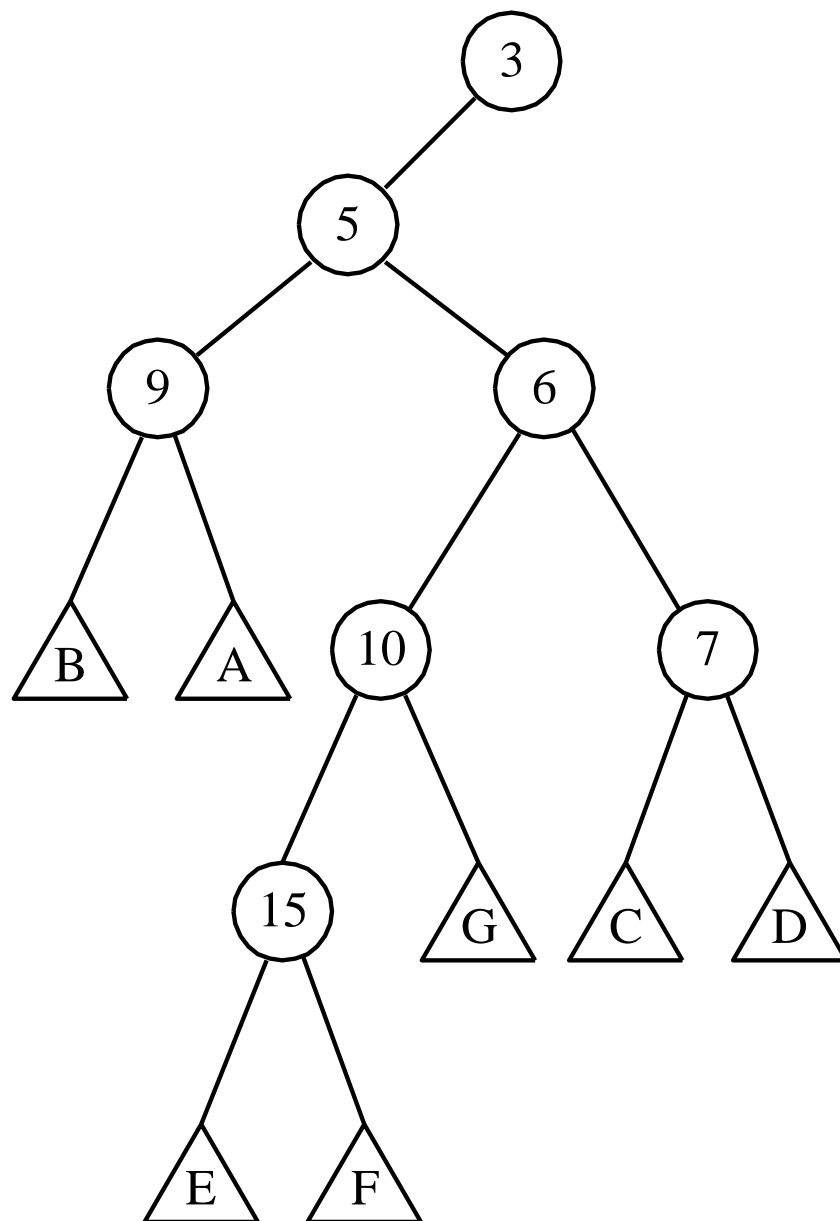
Krok 3: Postupné spájanie hálď do poslednej haldy





(c)

Binárna reprezentácia výslednej haldy:



Materiál slúži výlučne pre študentov FRI ŽU, nie je dovolené ho upravovať, prípadne ďalej šíriť.

2. Mnoho-prechodová párovacia halda (Multi pass pairing heap)

Krok 1: Výber koreňa (minimálny prvok) – výsledkom je niekoľko hald

Krok 2: Vloženie hald do FIFO frontu

Krok 3: Výber dvoch hald z frontu a ich spojenie do novej haldy, ktorú vložíme na koniec frontu

Krok 4: Ak je vo fronte len jedna halda - Koniec, inak chod' na Krok 3

Operácia Zvýšenia priority (vo viaccestnej forme) - priemerná aj najhoršia zložitosť $O(1)$:

V prípade, že prvok, ktorému sa zvýšila priorita je koreňom haldy, alebo jeho otec má vyššiu prioritu nie sú potrebné žiadne ďalšie úpravy. V opačnom prípade so stromu odpojíme prvok so zmenenou prioritou aj s jeho podstromami a spárujeme ho s koreňom haldy.

Operácia Zníženia priority (vo viaccestnej forme) - priemerná zložitosť $O(\log n)$, najhoršia zložitosť $O(n)$:

V prípade, že sa znížením priority narušila vlastnosť haldy (priorita otca je vyššia) je potrebné prvok so zmenenou prioritou z haldy odstihnúť spolu s jeho synmi (toto miesto je označené). Následne sa z každého syna stane samostatná halda, teda sa oddelí od prvku so zmenenou prioritou. Takto získaný les je následne rovnakým postupom ako pri mazaní prvku pomocou operácie párovania spojený do jedinej haldy. Táto halda sa pripojí do pôvodnej haldy na označené miesto, kde bol podstrom odobraný.

V zložitosti nie je započítané získanie (nájdenie) samotného prvku.

Operácia vymaž prvok - priemerná zložitosť $O(\log n)$, najhoršia zložitosť $O(n)$:

- zvýš prioritu odoberaného prvku na maximálnu možnú (napr. $-\infty$),
- aplikuj operáciu **Vymaž minimum**
- **v zložitosti nie je započítané získanie (nájdenie) samotného prvku**

Párovacia halda ponúka rovnakú efektivitu základných operácií (vlož a vymaž minimum) ako Fibonacciho halda, avšak implementácia príslušných algoritmov je značne jednoduchšia.