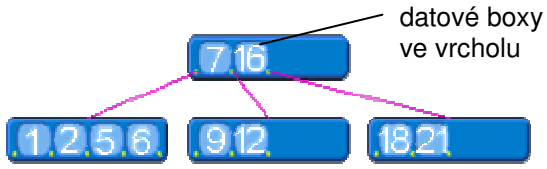
B- stromy



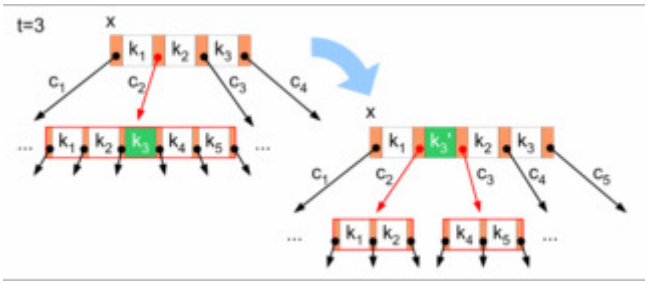
Jeho zvláštností je, že zavádí limity na maximální (konstantou 2k+1), i minimální (k) počet potomků vrcholu. B- strom je díky této vlastnosti vyvážený, operace přidání, vyjmutí i vyhledávání tedy probíhají v logaritmickém čase.
Tato struktura je často používána v aplikacích, kdy není celá struktura uložena v paměti RAM, ale v nějaké sekundární paměti, jako je pevný disk (například databáze).

Pravidla

- ne všechny vrcholy mají stejný počet synů
- počet datových boxů v uzlu je alespoň k a nejvýše 2k+1 pro nějaké k>1
- kořen datových boxů alespoň 1 a nejvýše 2k+1 (kořen stupně alespoň 2)
- všechny listy ve stejné hladině

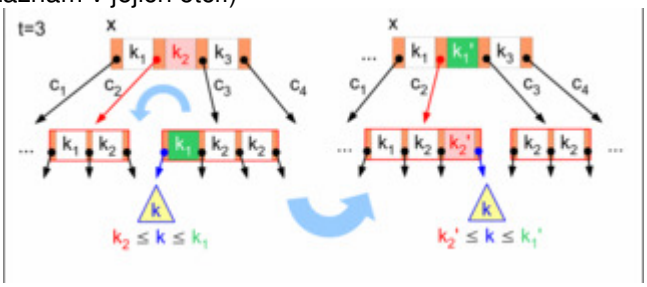
Insert

- pokud v uzlu U je ještě místo (je tam méně než 2k+1 záznamů), záznam tam přidáme (některé záznamy v U se přitom posunou, aby se nový záznam zařadil na správné místo podle klíče)
- pokud v uzlu U není místo (**je tam už 2k+1** záznamů), uzel U se rozdělí:
  * nový záznam zařadíme podle klíče mezi záznamy z uzlu U
  * prostřední z těchto záznamů označíme U
  * uzel U **rozštěpíme na dva** – do jednoho dáme záznamy s klíčem menším než U, do druhého s klíčem větším než U
  * záznam U i s ukazateli na oba nové uzly přidáme do otce bývalého uzlu U na místo, kde dosud byl ukazatel na uzel U … je-li tam místo, prostě přidáme, není-li místo, rozštěpíme ho a stejným způsobem postupujeme stromem nahoru
  → v důsledku toho může dojít až k rozštěpení dosavadního kořenu, vytvoření nového kořenu a tím ke zvýšení výšky stromu

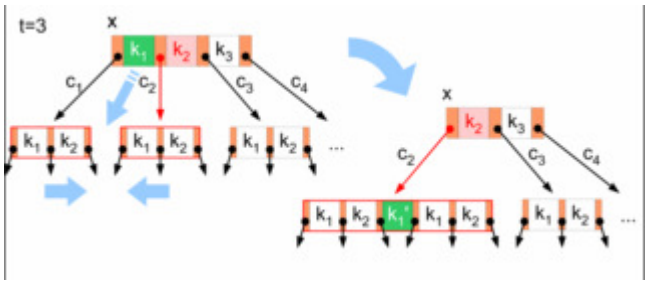


Delete

- vyhledat záznam podle klíče → uzel U, klíč k_i
- když U není list, existují jeho podstromy S_i, S_{i+1} → jako při vypouštění z BVS nahradíme vypouštěný záznam buď záznamem s maximálním klíčem z podstromu S_i nebo záznamem s minimálním klíčem z podstromu S_{i+1} a tam ho zrušíme (leží jistě v listu)
- dále tedy stačí řešit vypuštění záznamu z listu (nechť U je list):
- pokud je U zároveň také kořen, záznam z něj prostě vypustíme
- pokud U není kořen a obsahuje více než k záznamů, rušený záznam z něj prostě vypustíme
- pokud U není kořen a obsahuje pouze k záznamů, vypustíme rušený záznam a:
  * pokud U má levého či pravého „bratra“ s více než k záznamy, uzel U si od něj jeden záznam **převezme** (pozor – je nutná záměna přes „oddělující“ záznam v jejich otci!)



  * v opačném případě musí mít uzel U aspoň jednoho bratra s k záznamy → U se s ním **spojí** do jediného uzlu, který bude obsahovat zbylých k záznamů z U, všechny záznamy z bratra a jeden „oddělující“ záznam z jejich otce = nejvýše 2k+1 záznamů celkem, jak je povoleno při vypouštění oddělujícího záznamu z otce se může stát, že pak v otci zbude málo záznamů a uvedený postup se musí zopakovat (třeba i vícekrát, v krajním případě až ke zrušení kořenu a snížení výšky stromu)


t=3
x
k1
k2
k3
c1
c2
c3
c4
k1
k2
k1
k2
k1
k2
...
x
k2
k3
c2
c3
c4
k1
k2
k1
k1
k2
k1
k2
...