

Statický hešovací súbor

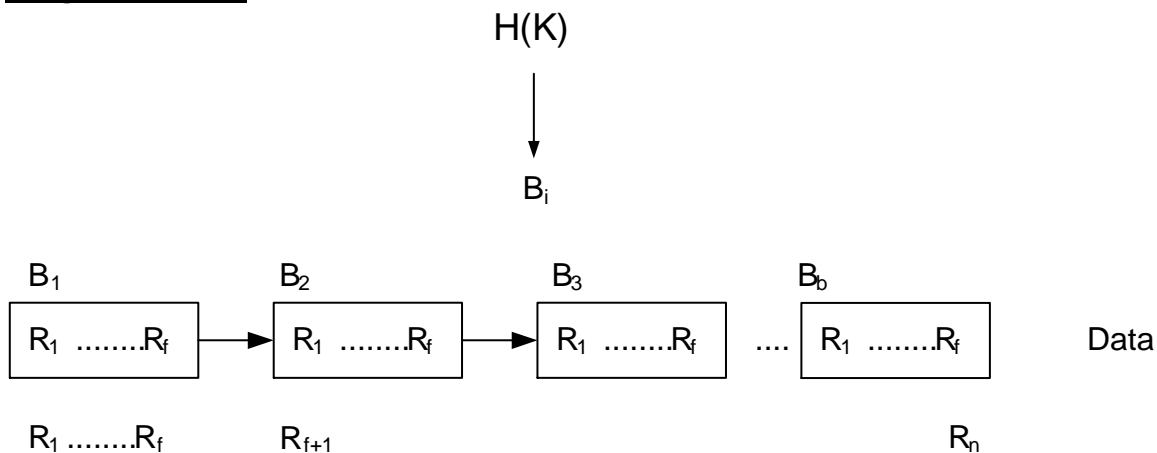
(Hash File)

- neutriedený súbor, prístup pomocou transformácie kľúča
- nie prístup k záznamu ale k skupine záznamov (k bloku)
- princíp je rovnaký ako hešovanie v operačnej pamäti (pozri podrobný popis z materiálu o hešovaní v operačnej pamäti)

Hlavný rozdiel od implementácie v operačnej pamäti je v adresovaní bloku:

$H(K) \longrightarrow$ adresa bloku

Organizácia:



Kolízia: až keď blok plný

Prístup:

- priamy podľa kľúča (výpočet adresy v adresári; link na blok)
- sériový (len ak sú bloky zreťazené)

Hešovacia funkcia:

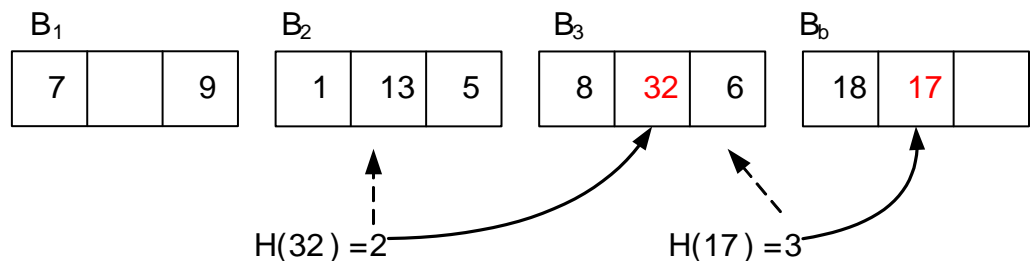
Namiesto mapovania do oblasti adries a_z záznamov $\langle 1 \dots n \rangle$ sa mapuje do oblasti adries a_b blokov $\langle 1 \dots n \div b \rangle$

Sprístupňovanie v bloku: sériovo (neutriedené).

Riešenie kolízií:

- a) Otvorené adresovanie
- b) Oblasť preplňujúcich blokov
- c) Preplňujúci súbor

a) Otvorené adresovanie:

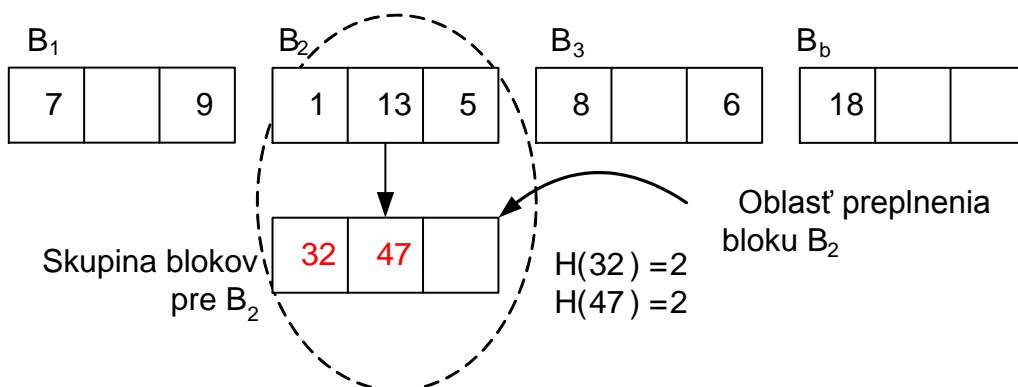


Hľadanie: po voľné miesto

Problém: rušenie, zhlukovanie (ako v operačnej pamäti – pozri príslušný materiál)

- ⇒ najmenej efektívne riešenie kolízií
- ⇒ všetky dáta sú v jednom súbore
- ⇒ so zaplnením prudko klesá efektivita vyhľadávania

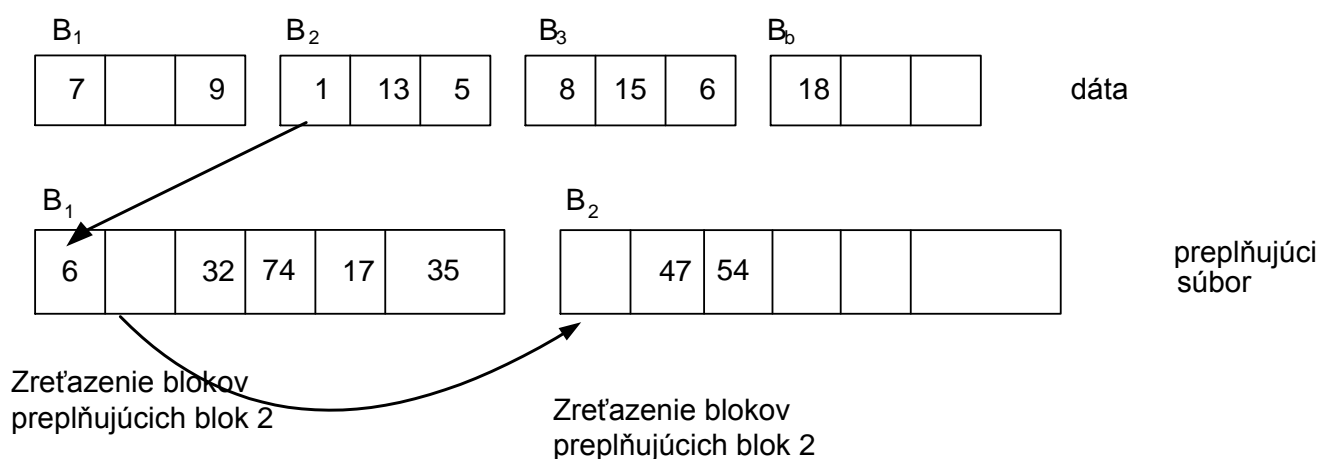
b) Oblasť preplňujúcich blokov:



- pri kolízii a prepĺnení bloku sa alokuje nový blok skupiny
- nový blok skupiny je alokovaný na konci hešovacieho súboru (ak nie je v oblasti prepĺnenia blokov žiadny voľný), alebo je pridelený voľný blok z oblasti prepĺnenia blokov (vysvetlené ďalej)
- adresuje sa skupina blokov

- časovo efektívnejšie
- pamäťovo menej efektívne (oblasti preplnenia málo zaplnené)
- v každom bloku si pamätáme index preplňujúceho bloku (prípadne presné miesto v súbore kde blok začína)
- výhodou je, že sú všetky dáta v jedinom súbore
- preplňujúce bloky sú taktiež zreťazené – dáta je možné vkladať až do zaplnenia disku
- štandardne sa preplňujúce bloky nachádzajú na konci súboru

c) Preplňujúci súbor (preplňujúce bloky sú uložené v osobitnom súbore):



snaha: bloky preplňujúceho súboru radšej väčšie, najlepšie počas spracovania v operačnej pamäti (v prípade, že je to možné)

- vlastnosti sú rovnaké ako pri využití oblasti preplňujúcich blokov
- pri preplnení sa alokuje nový blok skupiny blokov
- výhodou je možnosť alokovať preplňujúci súbor na iný disk

Práca s blokmi v preplňovacom súbore, alebo v oblasti preplňujúcich blokov:

- súbor s priamym prístupom je možné skrátiť iba od konca súboru
- ak sa uvoľní blok uprostred súboru nie je možné ho uvoľniť
- ak sa miesto uvoľní na konci súboru je súbor skrátený, ak ale pred uvoľňovaným blokom sú ďalšie voľné bloky musí sa súbor skrátiť aj o ne (situácia, keď sa obsadený blok nachádza na konci súboru a bráni skráteniu súboru o voľné bloky pred ním)
- ak je požadovaný nový voľný blok v preplňujúcom súbore, alebo oblasti preplňujúcich blokov je nutné uprednostniť existujúce voľné bloky pred zbytočným zväčšovaním súboru
- ak je možné viaceré zret'azené bloky zlúčiť, tak sa to uskutoční ihneď po vymazaní záznamu
- Na správu voľných blokov je možné využiť dva základné postupy:
 - a.) indexy (prípadne adresy) voľných blokov sa udržiavajú v operačnej pamäti
 - pri ukončení programu sa musia tieto indexy uložiť (buď do riadiaceho bloku súboru, prípadne do iného súboru)
 - b.) indexy (prípadne adresy) voľných blokov sa zret'azujú – každý blok má index (prípadne adresu) na nasledujúci blok
 - takto vzniká zret'azenie voľných blokov
 - nový uvoľnený blok, ktorý nie je možné uvoľniť skrátením súboru je pridaný na začiatok zret'azenia – maximálne jeden blokový prenos
 - pri požiadavke na nový voľný blok je pridelený prvý blok so zret'azenia – max. jeden blokový prenos
 - v operačnej pamäti je potrebné uchovávať index (adresu) prvého voľného bloku so zret'azenia, ktorá sa pri ukončení programu musí niekde uložiť (buď do riadiaceho bloku súboru, prípadne do iného súboru)

Záver o statickom hešovacom súbore:

- bežne 1-2 prenosi na sprístupnenie
- možnosť fixovania záznamov (nepremiestňujú sa)

- **nevýhody: nutná znalosť vlastností kľúčov, design na mieru, ťažká (niekedy nemožná) reorganizácia pri degenerácii**
- **nepoužívať pri silne dynamických súboroch**
- **plytvanie miestom na disku – statický hešovací súbor sa celý alokuje na začiatku a zaberá rovnaké miesto na disku bez ohľadu na jeho využitie (v prípade využitia oblasti preplňujúcich blokov môže veľkosť narastať)**