

9. Řízení vnější paměti

- K čemu slouží vnější paměť
- Charakteristika HDD
- Metody přidělování místa na disku
 - Spojité
 - Spojitý seznam
 - Indexová alokace
- Plánovací metody přístupu na disk
 - FCFS
 - SSTF
 - SCAN
 - C-SCAN
 - LOOK
 - C-LOOK
 - HDD vs. SSD
 - ◆ Defragmentace

1. K čemu slouží vnější paměť

- Vnější paměť slouží k ukládání informací(programy a data)
- Obsah ve vnější paměti se po vypnutí napájení neztratí(narozdíl od RAM)
- Procesor nemá přímý přístup k disku
- OS používá k přístupu do vnější paměti ovladače
- Data jsou organizována do souboru na základě souborového systému, který je použit
- Výhody
 - Nízké náklady
 - Energetická nezávislost
 - Nedestruktivní čtení
 - ◆ Přechtení informace ji nesmaže(narozdíl od RAM(kapacita kondiku))
- Stálá paměť: SSD, HDD
- Výměnná: disketa, flash, SD karta, CD, DVD

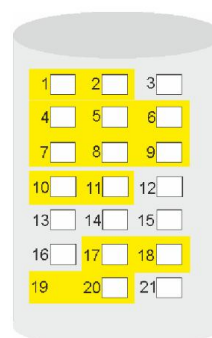
2. Charakterisika HDD

- Hard Disk Drive
- Data se uchovávají pomocí magnetické indukce
 - Hysterézní smyčka
- Výhodný poměr cena/objem dat
- Výhodou taky delší životnost než u SSD(pokud s diskem zacházíme šetrně, neházíme)
- Samotný disk je vyroben z nemagnetického materiálu
 - Jeho povrch je pokryt vrstvou ferromagnetického materiálu(oxid železa)

3. Metody přidělování místa na disku

● Spojité

- Nejjednodušší způsob
- Souvislá alokace, kde každý soubor zabírá několik bloků, které jdou po sobě
 - ◆ Přesně vím kde soubor začíná a kolik místa v paměti zabere
- Pokud je nový soubor moc velký, je nutná defragmentace
 - ◆ Tím vznikne větší počet po sobě jdoucích volných bloků
- Přístupy na disk
 - ◆ Sekvenční přístup
 - Před zpřístupněním informace z paměti je nutné přečíst všechny předcházející informace
 - Postupně se prochází od začátku místo v paměti
 - ◆ Přímý přístup
 - Je možné zpřístupnit přímo požadovanou informaci
 - Víme kde soubor začíná a kolik zabere->uloží rovnou na dané místo v paměti
- Výhoda->bloky na sebe navazují->malé pohyby hlaviček(vystavovacího mechanismu)
- Problém při vznikání nových souborů->dopředu nevíme kolik místa zaberou
- Pro přidělování volného místa se používají alokační strategie(algoritmy)
 - ◆ FIRST FIT



Metody přidělování místa na disku

1. Spojité (contiguous allocation)

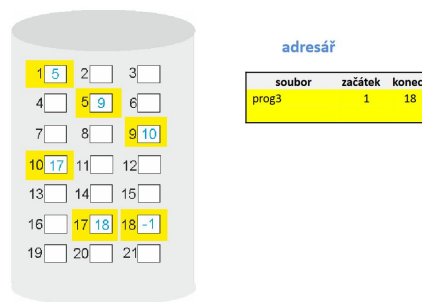
adresář

soubor	začátek	délka
s1.txt	1	2
prog1.c	4	6
seznam1.dat	10	2
prog2	17	4

- Nejjednodušší na implementaci
- ◆ BEST FIT
 - Vznik co nejmenší fragmentace
- ◆ LAST FIT
- ◆ WORST FIT
 - Uloží do největšího volného místa

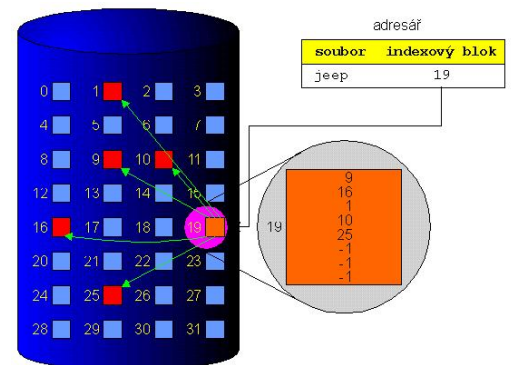
● Spojitý seznam

- Odstraňuje nutnost ukládat do bloků po sobě jdoucích
 - ◆ Stačí nám znát začátek a konec souboru (EOF)
- Soubor uložen do volných bloků tak, aby byly jednotlivé části souboru co nejbližší u sebe pro minimalizování pohybu hlaviček
 - ◆ Nepřiděluje se prázdné bloky jako rezerva pro růst souboru
 - ◆ Ovšem většinou pomalejší čtení(málokdy jsou bloky hned vedle sebe)
- Přístup pouze sekvenční-na konci bloku vždy ukazatel na další blok
- Není nutné znát velikost->nepřidávání bloků na víc->potlačení externí fragmentace
 - ◆ Po přidání FAT(File Allocation Table(na začátku disku)) je umožněn i přímý přístup
 - tabulka nám ukazuje na jednotlivé bloky
- Velikost alokačního bloku je dána v závislosti na použitém systému a kapacitě disku
 - ◆ Alokační blok je cluster
 - ◆ Příliš velké bloky->zbytek poslední nezaplněn úplně->Interní fragmentace
 - ◆ Využití v MS DOS a WIN95/96
 - 2. Spojitý seznam (linked allocation)



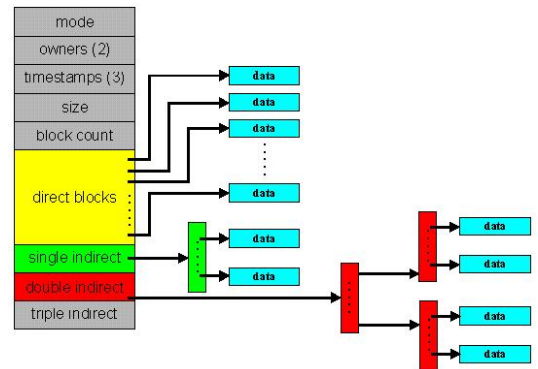
● Indexová alokace

- Indexy všech bloků souborů jsou umístěny pohromadě v indexovém bloku
 - ◆ Každý soubor má svůj indexový blok
 - ◆ Při poškození ztratíme pouze data souboru, a ne všechny data jako u FAT
- Přímý i sekvenční přístup
- Při práci se souborem je indexový blok nahrán do operační paměti
- Je složitější na realizaci, ale rychlejší
- Využití v UNIXu
- Vzniká vnitřní fragmentace(blok není využitý celý)
- Potlačená externí fragmentace
- Opět snaha soubor umístit co nejvíc u sebe
- Snaha o co nejmenší indexový blok
 - ◆ Metadata souboru
 - Atribut
 - Read
 - Write
 - Execute
 - Directory
 - Znakový speciální soubor
 - Blokový speciální soubor
 - Vlastník
 - Uživatel-vlastník
 - Skupina - možnost nastavení práv pro



jednotlivé skupiny abychom je mohli požit při přenosu a nemuseli je nastavova u každého jednoho uživatele

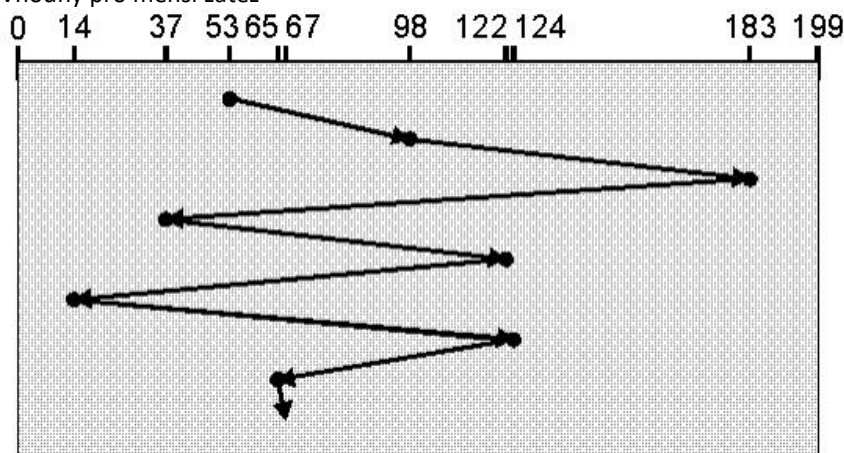
- Čas
 - Časové razítke - time stamps
 - Datum vytvoření, poslední změny a posledního přístupu
- Velikost
 - Aby se nemusela pokaždé načítat
- Počítadlo bloků
 - Kontrolní součet konečných dat, abychom věděli, jestli nějaké nechybí
 - Z pravidla 12 přímých bloků po 4kB, v případě že se nevejde, použijeme nepřímé bloky, kterých je taky 12, a ty odkazují na dalších 12,...



4. Plánovací metody přístupu na disk

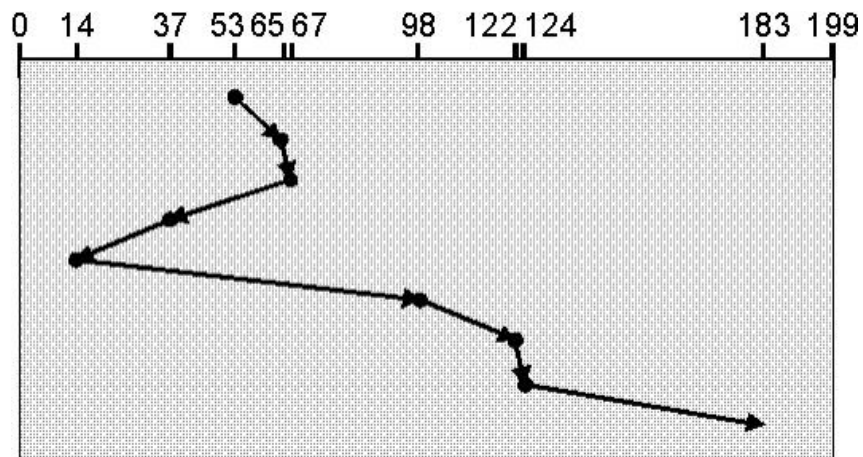
● FCFS

- First Come First Serve
- Fronta, kdo dřív přijde ten je dřív obsloužen
- Nejpomalejší
- Jednoduchý na programování
- Vhodný pro menší zátěž



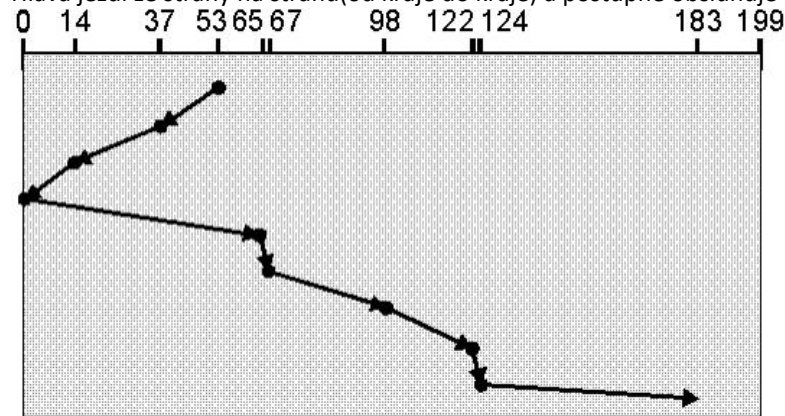
● SSTF

- Shortest Seek Time First
- Požadavek, který je nejbližší hlavičkám má přednost a bude obsloužen jako první
- Hrozí hladovění požadavků dál od hlavy
- Není ideální
- Rychlá metoda pro krátké vzdálenosti



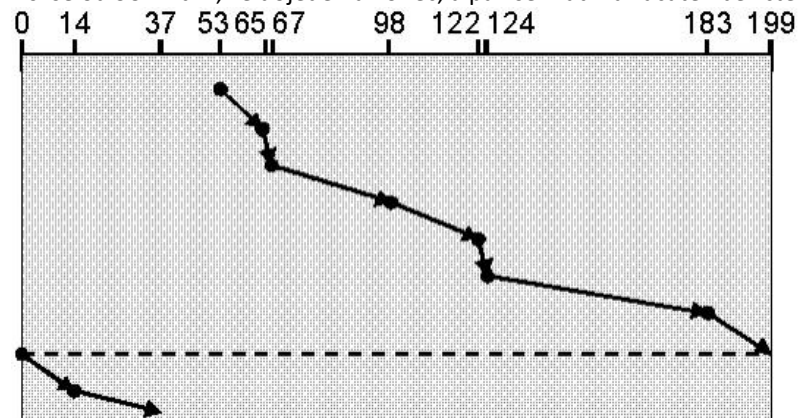
● SCAN

- Hlava jezdí ze strany na stranu (od kraje do kraje) a postupně obsluhuje požadavky



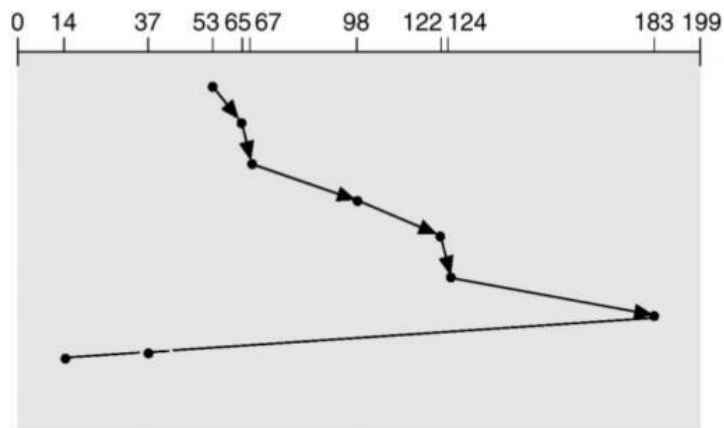
● C-SCAN

- Liší se od SCAN tím, že dojede na konec, a pak se vrátí na začátek bez čtení cestou zpět



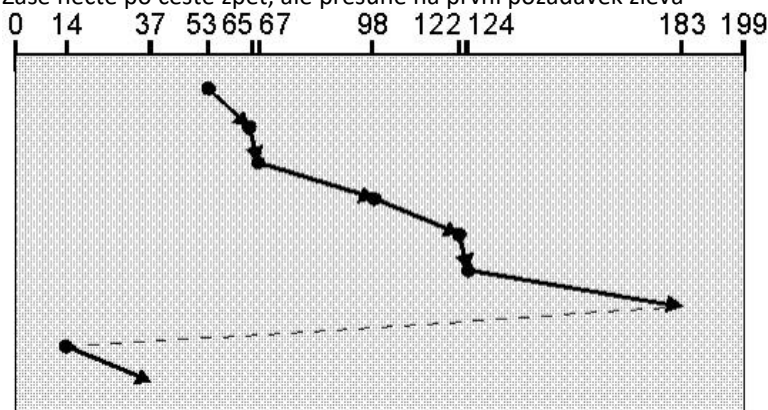
● LOOK

- Upravená metoda SCAN
- Hlava nedojíždí až ke kraji, dojede pouze k poslednímu požadavku a pak se vrací zpět



● C-LOOK

- Vylepšená LOOK
- Zase nečte po cestě zpět, ale přesune na první požadavek zleva



5. HDD vs. SSD

● HDD(Hard Disk Drive)

- Větší přístupový čas->pomalejší
- Větší spotřeba elektřiny kvůli pohyblivým součástkám
- Dochází k fragmentaci dat->zhoršení výkonu
- Hrozí poruchy z důvodu mechanických částí disku-plotny, hlavy, vystavovací mechanismus
- Magnetická vrstva na povrchu se může při vibracích/otřesech poškodit
- Hlučnější

● SSD(Solid State Drive)

- Kratší přístupový čas->rychlejší
- Žadné mechanické součástky->odolnější na poškození
- Nedochází k fragmentaci
- Jedná se o integrovaný obvod
- Je dražší
- Většinou menší

● Defragmentace

- Defragmentace je proces optimalizace dat na disku tím, že fyzicky uspořádá fragmentované soubory tak, aby byly uloženy na disku kontinuálně. To pomáhá zvýšit rychlost přístupu k datům a celkový výkon disku. U SSD není defragmentace tak kritická jako u HDD, protože SSD nemá mechanické části a přístupové časy jsou konzistentní bez ohledu na to kde jsou data uložena
- Defragmentační programy
 - ◆ V OS je to defragmentace
 - ◆ O&O Defrag
 - ◆ Diskkeeper